

What is Scriptcase?

Scriptcase is a complete development tool, with a friendly web interface. A low-code environment that creates PHP applications high-speed and with quality was built to save time, reduce costs, gain productivity, and to integrate the development teams. Using Scriptcase, you can create many types of applications like **Grids, Charts, Forms, Calendars, and many other applications** . It also offers an IDE integrated for you to customize applications or even to create using only codes.

You can divide Scriptcase environment in 3 parts: **RAD (Rapid Application Development)**, **Scriptcase environment tools** and **Code editors**.

- **AD (Rapid Application Development)**: Within Scriptcase, you can find some ready to use applications as well as complete system templates. You can see a sample of all applications accessing our website [here](#).
- **criptcase environment tools**: You have available plenty of tools to assist your development within Scriptcase, such as Database Manager, SQL Builder, Data dictionary, CSV, XLS and Access converters, Documentation Creator, Themes Manager, Images Manager, Libraries Manager, Tasks Manager, and Key Shortcuts.
- **ode editors**: Besides the ready to use applications, templates, and tools, you can also customize or create your application using the editors, they work like an IDE integrate to Scriptcase. Within the editors, you can add your system rules integrating with web services, third-party libraries and components, APIs, and Scriptcase internal functions (macros). The editors are compatible with PHP, JavaScript, CSS, and HTML, and you can also use Emmet Shortcuts.

How does Scriptcase work?

Scriptcase works generating web applications in PHP and always relating the content with the most used databases on the market. Installing on a web server, Scriptcase can be accessed simultaneously by many developers through the browser, allowing remote and collaborative development. The source code is in PHP, and it works independently of the tool. Once you finish your project on Scriptcase, you can deploy it anywhere with a WEB server that allows you work with PHP. You do not need a license to run the generated applications. Scriptcase provides a "Production environment" where you can manage the libraries, connections, and deployed applications. Learn more about the deployment [here](#).

Which source code is generated?

Scriptcase generates (PHP, JavaScript, and HTML) source code. The generated code is independent of the tool, and you can deploy it on a web server with PHP enabled. The PHP language is free (GNU), and you can use on any operating system (Windows, Linux, or Mac). PHP processes work on the Server.

What are the supported databases?

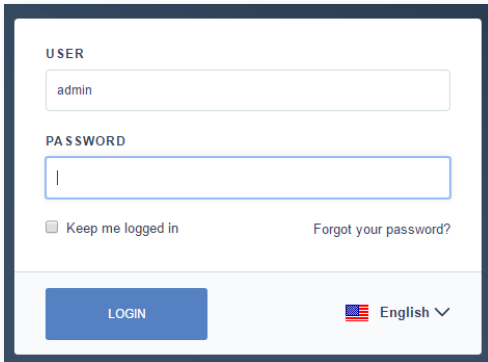
Scriptcase connects to the leading relational databases: Oracle, MS-SQL Server, DB2, MySQL, Informix, PostgreSQL, Access, InterBase, FireBird, Progress, and others ODBC or OLEDB patterns.

What are the requirements to run ScriptCase?

A webserver (Apache, IIS, Sambar, Abyss, PWS, Xitami) that is enabled to run PHP 7.3 Scripts and IonCube Loader.

When accessing ScriptCase, you find the login screen where you enter the username and password to log on.

For your first access, the username and password is **admin**. We recommend you to change the default password by accessing the environment settings panel after the first login.




USER

admin

PASSWORD

Keep me logged in [Forgot your password?](#)

LOGIN  English ▾

After validating the username and password, you access all of the Scriptcase available functions. The “Administrator” user has privileged access to all the ScriptCase functionalities and is also able to create new users to use the same installation concurrently. The simultaneous use depends on the license arrangement you have.

If you have forgotten your password and need to recovery, you must have the [SMTP](#) server configured within for a successful process. Check [here](#) a tutorial showing how to make the recovery.

- criptCase installer includes PHP 7.3, Apache 2.4, and IonCube Loader.
- he installation process is described according to the Operating System.

Installing ScriptCase is not complicated, select your operating system below, follow the tutorial, and in a few steps, you get to install and start using ScriptCase. If you need any assistance, please contact our technical support [here](#).



The Scriptcase Installer is an executable file for Windows. An application for automatic extraction and configuration of **Apache, PHP and Scriptcase** on your operating system. **The Installer will leave the environment ready, with all the correct values for Scriptcase use.**

WARNING: The Scriptcase Installer is compressed into a **.zip** file, you will need to use a software of your own to decompress the **.exe** file and run it.

Prerequisites

To download the Scriptcase Installer, go to our download page clicking [here](#).

Listed below are all the supported operating systems, as well as the libraries required for operation on a particular system.

Check the list of supported operating systems:

- indows 7 SP1 or Higher
- indows Vista SP2 or Higher
- indows Server 2008 SP2 or Higher
- indows Server 2008 R2 SP1 or Higher

For **Windows 7**, you must verify if the system has the following items installed:

- isual C++ Redistributable Packages for Visual Studio (2008, 2010, 2012, 2013, 2015) **x86** and **x64**.

NOTE: Before proceeding with the installation, it is recommended not to have any other web environments installed (such as WampServer, XAMPP, Zend Server, etc.) on your operating system.

This documentation will describe Scriptcase installation in 2 simple steps:

- 1 - Choose installer settings.
- 2 - Run Scriptcase.

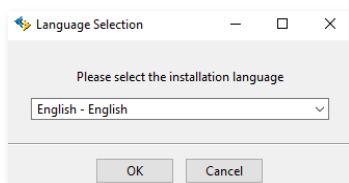
Choosing settings in the Installer

NOTE: Installing Scriptcase with PHP 7.3 will not conflict with an existing Scriptcase installation with PHP 7.0. The Installer configures an Apache service (**ApacheScriptcase9php73**) with a different port (**8091**) than the installer with PHP 7.0.

Will be automatically installed on the operating system:

- pache Web Server **v2.4.39**.
- HP **7.3** with extensions **MBSTRING, ZIP, BCMATH** and **GD** enabled.
- oncube: PHP Loader **v10.3.8**.
- criptcase **9**.

1 - Choose the installation language.

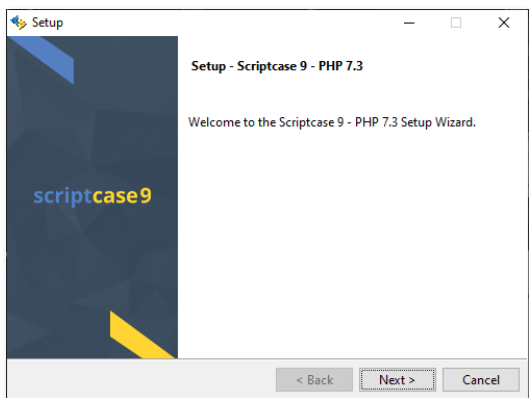


Language

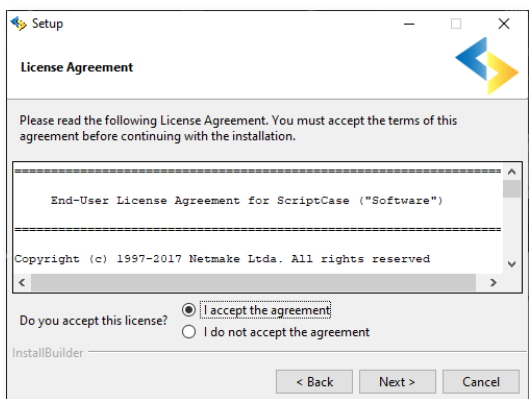
You can select the language that will be displayed in the installation process. This step does not select the Scriptcase interface language, only the installer language.

Click **Ok** to continue.

2 - Click **"Next"** to continue.



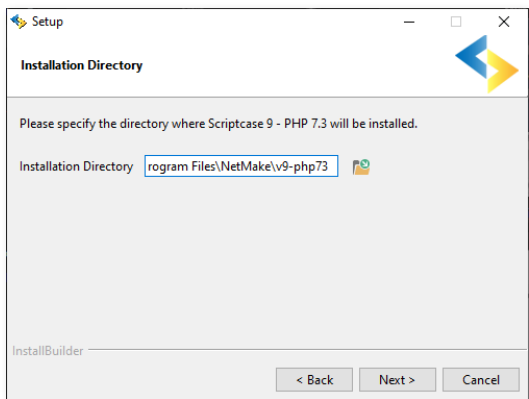
3 - Read and accept the terms of the license agreement.



Click **Next** to continue.

4 - Enter the installation directory.

The default installation path is "**C:\Program Files\NetMakev9-php73**". It is not recommended to make any changes to this path unless it is necessary.



Installation directory

In this option, you can set the installation path.

 : This button lets you select the installation folder or create a new one.

Click **Next** to continue.

5 - Configure Scriptcase Apache.

Port

Here will define in which port the Scriptcase Apache service will run on. We recommend using the value set in this installation. **(8091)**.

Admin Email

You can configure Apache administrator email for better control of your environment. This setting serves as information when a server error occurs. A message will appear along with the administrator email to be contacted.

Internal Server Error

The server encountered an internal error or misconfiguration and was unable to complete your request.

Please contact the server administrator at `root@localhost` to inform them of the time this error occurred, and the actions you performed just before this error.

More information about this error may be available in the server error log.

Apache Domain

The Apache domain must be set here. We do not recommend making any changes to the default value.

PHP TimeZone

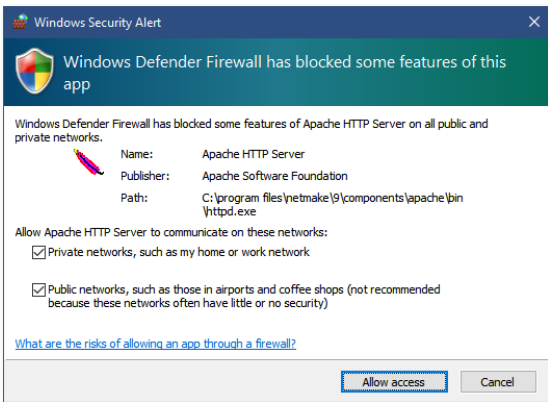
In this option, you can set your PHP TimeZone. It is recommended to select the option corresponding to your time zone. You can check the list of available TimeZones [here](#).

Click **Next** to continue.

6 - Clicking "Next" will start the installation process.

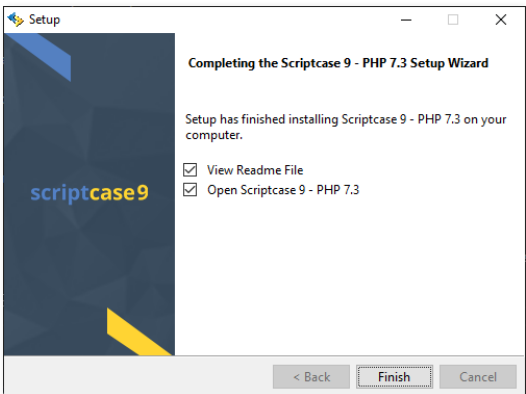
Wait until all components are installing and the environment is set up.

7 - Allow Apache to run on private and public networks (Firewall).



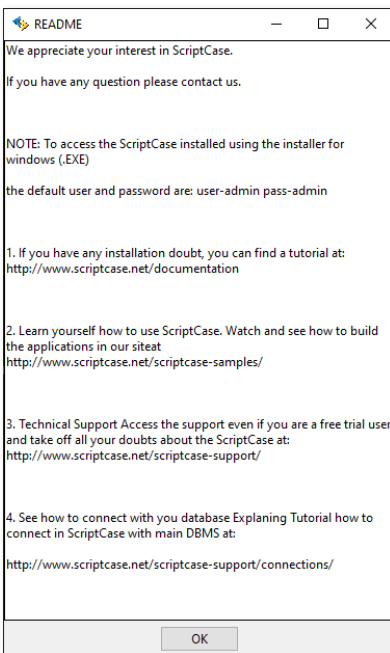
NOTE: You must check both options to allow Scriptcase Apache to run on the operating system.

8 - Complete the installation.



View the "Read Me" file

Clicking finish will show the "Read Me" file with all the necessary initial information and helpful links.



Open Scriptcase 9

Clicking "Finish", will open Scriptcase in your default browser.

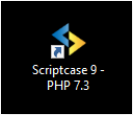
Access the Scriptcase

You can access your Scriptcase in two ways:

- 1 - Access your browser and enter the local IP address and port you selected in the Apache setup.

Example: <http://127.0.0.1:8091/scriptcase>

2 - After Scriptcase installation, run the shortcut created on your desktop. Just click on this shortcut twice and Scriptcase will open in your default browser.



Login to Scriptcase

After accessing the Login page, you must enter the username and password to proceed to your development environment.

A screenshot of the Scriptcase login page. The page has a dark blue background with the Scriptcase logo (two arrows pointing towards each other) and the text 'scriptcase' in white and yellow. Below the logo is a white login form with the following elements: a 'USER' label above a text input field; a 'PASSWORD' label above a text input field; a checkbox labeled 'Keep me logged in'; a link labeled 'Forgot your password?'; a blue 'LOGIN' button; and a language selector showing 'English' with a dropdown arrow. At the bottom of the form, there is a note: 'To access for the first time please use `admin / admin`. Your trial will expire in 20 days.'

User

In this field, you must enter the user to login to Scriptcase. By default, the user is **admin**.

Password

In this field, you must enter the password to login to Scriptcase. By default, the password is **admin**.

Keep me logged in

This option keeps the user logged in until they log out through the Scriptcase interface.

Forgot your password?

IMPORTANT: Password recovery will only be possible if you have configured your SMTP information. To access our documentation showing how to perform this configuration click [here](#).

By clicking on this option, it is possible to recover the password of the informed user. An email will be sent to the user so they can access the recovery steps.

Need to register your Scriptcase?

You can access our [knowledge base](#) to understand all the steps for registering a Scriptcase license.

The Scriptcase automatic installer is an executable file for Linux systems. An application for automatic extraction and configuration of **Apache, PHP and Scriptcase** on your operating system. **The installer will leave the entire environment ready with all the correct values for Scriptcase use.**

Prerequisites

To download the Scriptcase Installer, go to our [download page](#) clicking [here](#).

Listed below are the operating systems approved for Scriptcase use.

WARNING: Only the distributions below and derived distributions are approved for Scriptcase installation.

Check the list of supported operating systems:

- edHat\CentOS
- ebian\Ubuntu
- use\OpenSuse

NOTE: Before proceeding with the installation, it is recommended not to have any other web environments installed (such as XAMP, Zend Server, etc) on your operating system. **It is recommended to have some basic knowledge of Linux/Unix environment to use the Terminal (Shell).**

This documentation will describe Scriptcase installation in 3 simple steps:

- 1 - Run Scriptcase Installer.
- 2 - Choose installer settings.
- 3 - Run Scriptcase.

Running Scriptcase Installer

1 - Browse to the directory where the installer is located.

Before we can run the Scriptcase installer, we need to access the folder where the installer file is located using the `cd` command.

Example: `sudo cd /home/your_user/Downloads`

```
leonardo@Netmake-NM03:~$ cd /home/leonardo/Downloads/  
leonardo@Netmake-NM03:~/Downloads$ _
```

NOTE: Scriptcase installer requires administrative permissions to make changes to your machine. We will have to use the `sudo` command to run the installer.

2 - Grant permission to the installer file and run it.

For you to run the `chmod` command that changes the permissions of a file or directory, you must also run the `sudo` command.

Example: `sudo chmod +x scriptcase_linux_installer_x86_64_en_us_v9.4.016-php73.run`

```
~/Downloads$ sudo chmod +x scriptcase_linux_installer_x86_64_pt_br_v9.4.016-php73.run
```

The `chmod +x` command is used to make the installer `.run` file executable.

3 - Run the Installer file.

Example: `sudo ./scriptcase_linux_installer_x86_64_en_us_v9.4.016-php73.run`

```
~/Downloads$ sudo ./scriptcase_linux_installer_x86_64_pt_br_v9.4.016-php73.run
```

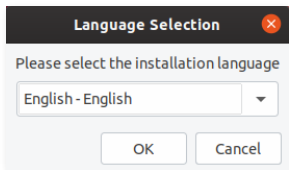
Choosing the Installer settings

NOTE: Installing Scriptcase with PHP 7.3 will not conflict with an existing Scriptcase installation with PHP 7.0. The Installer configures an Apache service (**apachesc9php73**) with a different port (**8091**) than the installer with PHP 7.0.

Will be automatically installed on the operating system:

- pache Web Server **v2.4.39**.
- HP **7.3** with extensions **MBSTRING, ZIP, BCMATH** and **GD** enabled.
- oncube: PHP Loader + ionCube24 **v10.3.7**.
- criptcase **9**.

1 - Choose the installation language.



Language

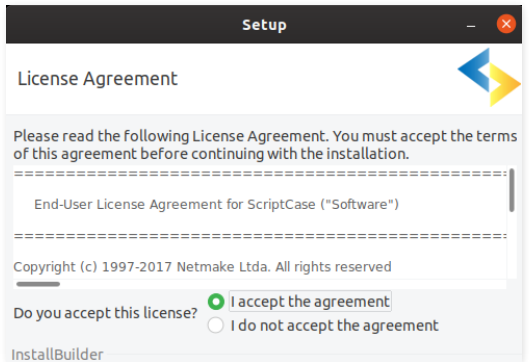
You can select the language that will be displayed in the installation process. This step does not select the Scriptcase interface language, only the installer language.

Click **OK** to continue

2 - Click "Next" to continue.



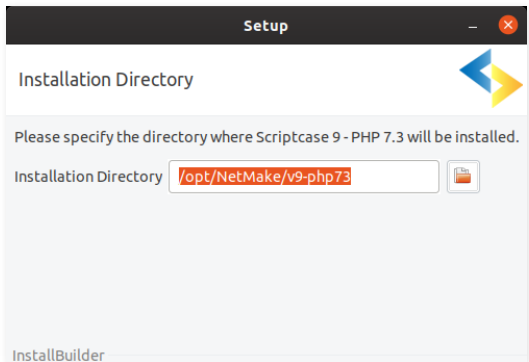
3 - Read and accept the terms of the license agreement.



Click **Next** to continue.


4 - Enter the installation directory.

The default installation path is `"/opt/NetMake/v9-php73"`. It is not recommended to make any changes to this path unless it is necessary.



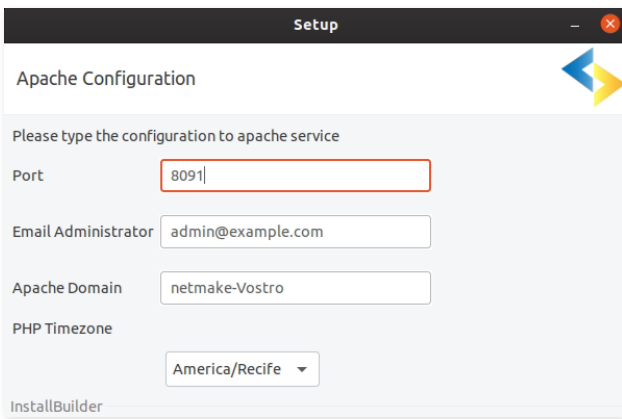
Installation Directory

In this option, you can set the installation path.

: This button lets you select the installation folder or create a new one.

Click **Next** to continue.

5 - Configure Scriptcase Apache.



Port

Here will define in which port the Scriptcase Apache service will run on. We recommend using the value set in this installation **(8091)**.

Admin Email

You can configure Apache administrator email for better control of your environment. This setting serves as information when a server error occurs. A message will appear along with the administrator email to be contacted.

Internal Server Error

The server encountered an internal error or misconfiguration and was unable to complete your request.

Please contact the server administrator at `root@localhost` to inform them of the time this error occurred, and the actions you performed just before this error.

More information about this error may be available in the server error log.

Apache Domain

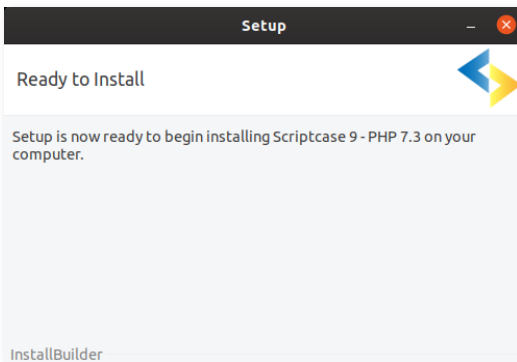
The Apache domain must be set here. We do not recommend making any changes to the default value.

PHP TimeZone

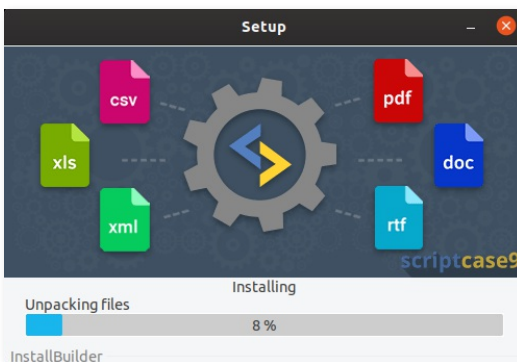
In this option, you can set your PHP TimeZone. It is recommended to select the option corresponding to your time zone. You can check the list of available TimeZones [here](#).

Click **Next** to continue.

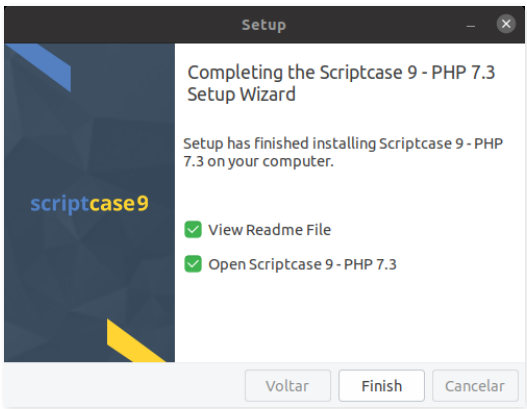
6 - Clicking "Next" will start the installation process.



Wait until all components are installing and the environment is set up.

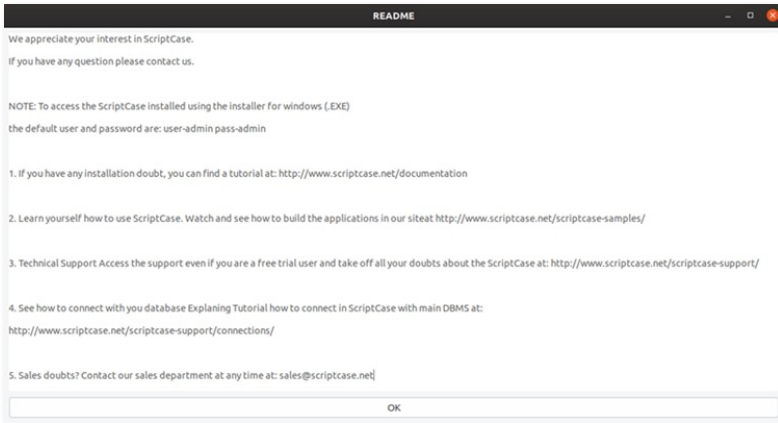


8 - Complete the installation.



View the "Read Me" file

Clicking finish will show the "Read Me" file with all the necessary initial information and helpful links.



Open Scriptcase 9

Clicking "Finish", will open Scriptcase in your default browser.

Access the Scriptcase

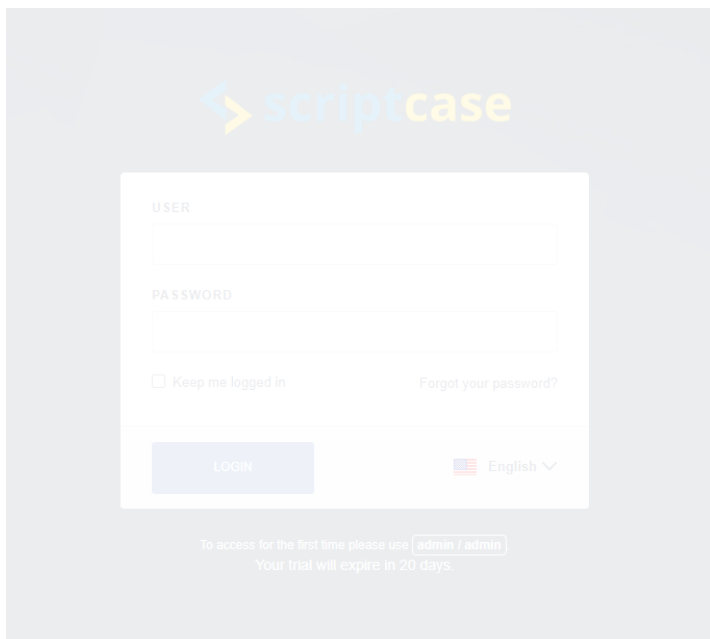
1 - To access your Scriptcase, you need to open your browser and enter the local IP or domain of the ported server you selected in the Apache setup.

Example 1: `http://127.0.0.1:8091/scriptcase`

Example 2: `https://yourdomain.com:8091/scriptcase`

2 - Login to Scriptcase

After accessing the Login page, you must enter the username and password to proceed to your development environment.



scriptcase

USER

PASSWORD

Keep me logged in [Forgot your password?](#)

LOGIN English ▾

To access for the first time please use | admin / admin |.
Your trial will expire in 20 days.

User

In this field, you must enter the user to login to Scriptcase. By default, the user is **admin**.

Password

In this field, you must enter the password to login to Scriptcase. By default, the password is **admin**.

Keep me logged in

This option keeps the user logged in until they log out through the Scriptcase interface.

Forgot your password?

IMPORTANT: Password recovery will only be possible if you have configured your SMTP information. To access our documentation showing how to perform this configuration click [here](#).

By clicking on this option, it is possible to recover the password of the informed user. An email will be sent to the user so they can access the recovery steps.

Need to register your Scriptcase?

You can access our [knowledge base](#) to understand all the steps for registering a Scriptcase license.

The Scriptcase automatic installer is an executable file for macOS. An application for automatic extraction and configuration of **Apache, PHP and Scriptcase** on your operating system. **The installer will leave the entire environment ready with all the correct values for Scriptcase use.**

IMPORTANT: The Scriptcase Installer is compressed into a **.zip** file, you will need to use a software of your own to decompress the **.app**. To run the file, you will need to click with the right button and after that, click in **Open** to authenticate and install Scriptcase.

Prerequisites

To download the Scriptcase Installer, go to our download page clicking [here](#).

Listed below are the macOS versions approved for Scriptcase use.

Check the list of supported versions:

- S X v10.11 "El Capitan"
- acOS v10.12 "Sierra"
- acOS v10.13 "High Sierra"
- acOS v10.14 "Mojave"
- acOS 10.15 "Catalina"

NOTE: Before proceeding with the installation, it is recommended not to have any other web environments installed (such as XAMP, Zend Server, etc) on your operating system.

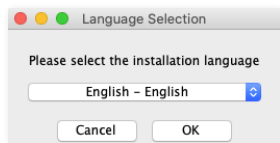
Choosing settings in the Installer

NOTE: Installing Scriptcase with PHP 7.3 will not conflict with an existing Scriptcase installation with PHP 7.0. The Installer configures an Apache service (**apachesc9php73**) with a different port (**8091**) than the installer with PHP 7.0.

Will be automatically installed on the operating system:

- pache Web Server **v2.4.41**.
- HP **7.3** with extensions **MBSTRING, ZIP, BCMATH** and **GD** enabled.
- oncube: PHP Loader **v10.3.8**.
- criptcase **9**.

1 - Choose the installation language.



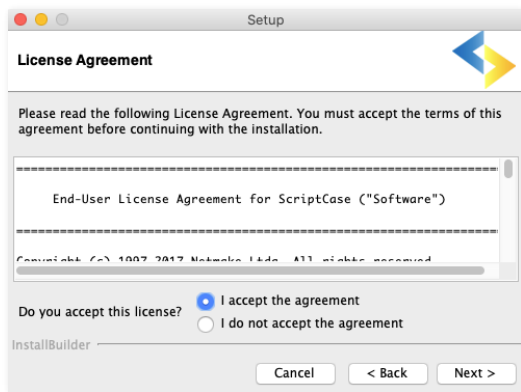
Language

You can select the language that will be displayed in the installation process. This step does not select the Scriptcase interface language, only the installer language.

Click **OK** to continue



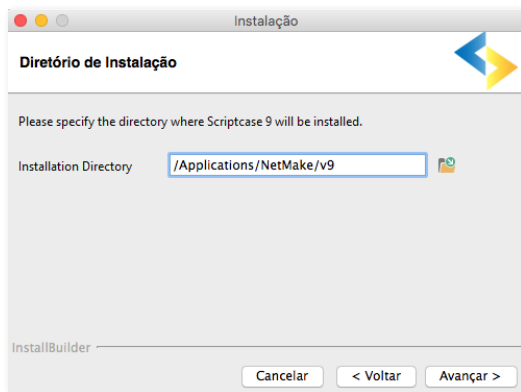
2 - Click "Next" to continue.



Click **Next** to continue.


4 - Enter the installation directory.

The default installation path is `"/Applications/NetMake/v9-php73"`. It is not recommended to make any changes to this path unless it is necessary.



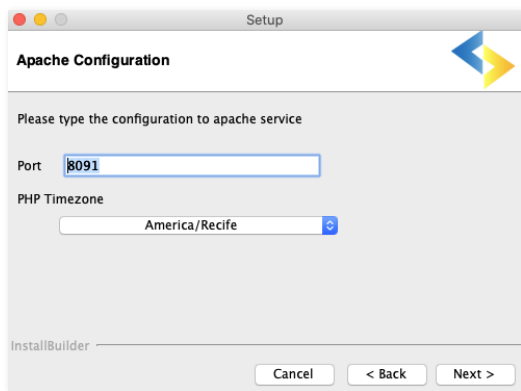
Installation directory

In this option, you can set the installation path.

 : This button lets you select the installation folder or create a new one.

Click **Next** to continue.

5 - Configure Scriptcase Apache.



Port

Here will define in which port the Scriptcase Apache service will run on. We recommend using the value set in this installation (**8091**).

Admin Email

You can configure Apache administrator email for better control of your environment. This setting serves as information when a server error occurs. A message will appear along with the administrator email to be contacted.

Internal Server Error

The server encountered an internal error or misconfiguration and was unable to complete your request.

Please contact the server administrator at `root@localhost` to inform them of the time this error occurred, and the actions you performed just before this error.

More information about this error may be available in the server error log.

Apache Domain

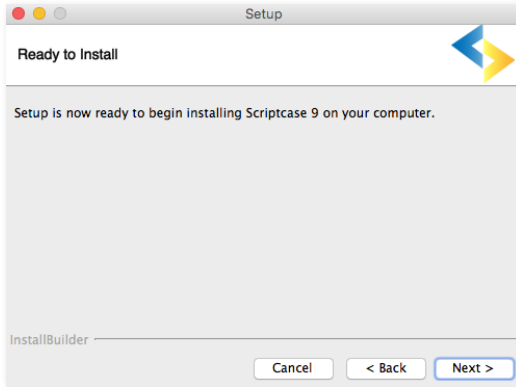
The Apache domain must be set here. We do not recommend making any changes to the default value.

PHP TimeZone

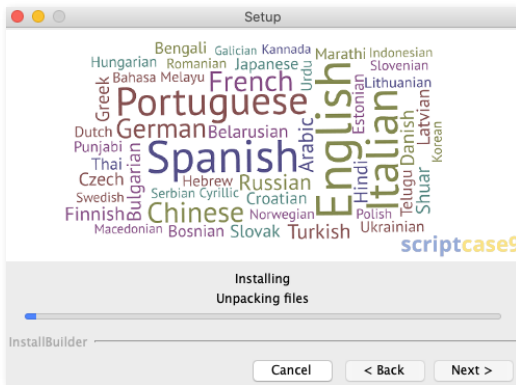
In this option, you can set your PHP TimeZone. It is recommended to select the option corresponding to your time zone. You can check the list of available TimeZones [here](#).

Click **Next** to continue.

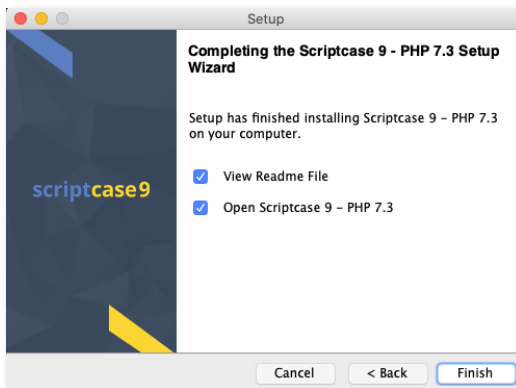
6 - Clicking "Next" will start the installation process.



Wait until all components are installing and the environment is set up.

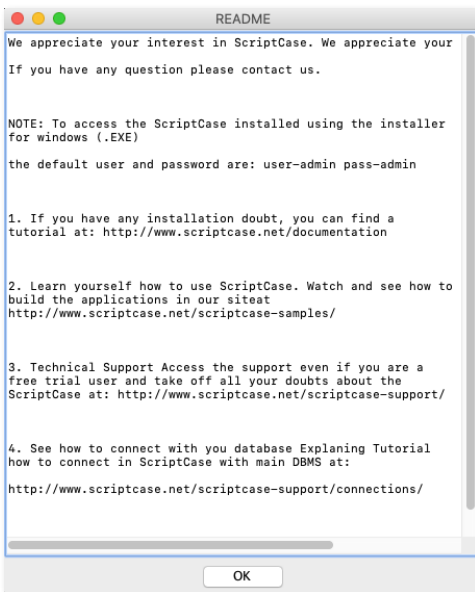


8 - Complete the installation.



View the "Read me" file

Clicking finish will show the **"Read Me"** file with all the necessary initial information and helpful links.



Open Scriptcase 9

Clicking finish **"Finish"** will open Scriptcase in your default browser.

Access Scriptcase

1 - To access your Scriptcase, you need to open your browser and enter the local IP or domain of the ported server you selected in the Apache setup

Example 1: <http://127.0.0.1:8091/scriptcase>

Example 2: <https://yourdomain.com:8091/scriptcase>

2 - Login to Scriptcase

After accessing the Login page, you must enter the username and password to proceed to your development environment.

User

In this field, you must enter the user to login to Scriptcase. By default, the user is **admin**.

Password

In this field, you must enter the password to login to Scriptcase. By default, the password is **admin**.

Keep me logged in

This option keeps the user logged in until they log out through the Scriptcase interface.

Forgot your password?

IMPORTANT: Password recovery will only be possible if you have configured your SMTP information. To access our documentation showing how to perform this configuration click [here](#).

By clicking on this option, it is possible to recover the password of the informed user. An email will be sent to the user so they can access the recovery steps.

Need to register your Scriptcase?

You can access our [knowledge base](#) to understand all the steps for registering a Scriptcase license.

This documentation will walk you through the steps necessary to install PHP 7.3, Apache 2.4, and Scriptcase manually. **When performing this type of installation, you are responsible for configuring your entire environment as well as database extensions.**

Listed below are all the supported operating systems as well as the libraries that are required to function on a particular system.

Check the list of supported operating systems:

- indows 7 SP1 or Higher
- indows Vista SP2 or Higher
- indows Server 2008 SP2 or Higher
- indows Server 2008 R2 SP1 or Higher

For **Windows 7**, you must verify if the system has the following items installed:

- isual C++ Redistributable Packages for Visual Studio (2008, 2010, 2012, 2013, 2015) **x86** and **x64**.

NOTE: If you would like to manually install Scriptcase with **PHP 7.0** on Windows, please access the documentation in our [knowledge base](#).

Prerequisites

WARNING: If you already have a Web Server with Apache and PHP 7.3 configured, just go to the [lonCube](#) configuration and proceed to manual installation.

To proceed with the installation, you need to download some files first. Those include the Scriptcase installation file in **.zip** format.

Required Files:

- HP 7.3 NTS (x86 or x64) : [Click Here](#)
- pache 2.4 (x86 or x64) : [Click Here](#)
- od_Fcgi (x86 or x64) : [Click Here](#)
- onCube Loader for **Windows VC15 (x86 or x64)** : [Click Here](#)
- criptcase (.zip): [Click Here](#)

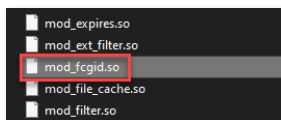
NOTE: Before proceeding with the installation, it is recommended not to have any other web environments installed (such as WampServer, XAMPP, Zend Server, etc.) on your operating system.

Below are the actions required after downloading the items specified above::

- 1 - Unzip the **PHP** and **Apache** files in separate folders on your operating system's C: drive.
- 2 - Rename the PHP folder to **php**. The result should be like this:

Name	Date modified	Type	Size
Apache24	16/03/2014 08:14	File folder	
PerfLogs	22/08/2013 12:22	File folder	
php	18/07/2014 11:53	File folder	
Program Files	18/07/2014 10:17	File folder	
Program Files (x86)	18/07/2014 10:47	File folder	
Users	17/07/2014 19:30	File folder	
webserv	18/07/2014 10:50	File folder	
Windows	17/07/2014 18:13	File folder	

- 3 - Unzip the **mod_Fcgi** file, copy the **mod_fcgid.so** file to the **C:\Apache24\modules** folder.



Configure the Apache environment

Listed below are the steps required to configure apache manually. **Follow the steps correctly to avoid problems installing and starting the service.**

NOTE: Since PHP and Apache folders are at the same directory level (**C:**) You do not need to enter the root folder in directives that require you to enter isolated paths or points to a specific file.

- 1 - Edit the **httpd.conf** file located in the **C:\Apache24\conf** folder using a text editor in Administrator mode, so that any changes you make are saved.

extra	09/08/201
original	09/08/201
charset.conv	09/08/201
httpd.conf	09/12/201
magic	09/08/201
mime.types	09/08/201

2 - Add the content below after the `#LoadModule xml2enc_module modules/mod_xml2enc.so` line:

```
LoadModule fcgid_module modules/mod_fcgid.so
FcgidInitialEnv PHPRC "/php"
FcgidInitialEnv PHP_FCGI_MAX_REQUESTS "100000"
FcgidIOTimeout 1800
FcgidBusyTimeout 1800
FcgidConnectTimeout 1800
FcgidMaxRequestLen 1073741824
FcgidMaxRequestsPerProcess 10000
FcgidOutputBufferSize 4000
FcgidProcessLifeTime 3600
Timeout 1024
KeepAlive On
KeepAliveTimeout 50
MaxKeepAliveRequests 500
AddHandler fcgid-script .php
FcgidWrapper "/php/php-cgi.exe" .php
```

Example:

```
#LoadModule xml2enc_module modules/mod_xml2enc.so
LoadModule fcgid_module modules/mod_fcgid.so
FcgidInitialEnv PHPRC "/php"
AddHandler fcgid-script .php
FcgidWrapper "/php/php-cgi.exe" .php
<IfModule unixd_module>
```

WARNING: This setting will only work if you have copied the `mod_fcgid.so` file to the `C:\Apache24\modules` folder as [previously instructed](#).

3 - Search for `#ServerName www.example.com:80` and change this line to `ServerName localhost:80`.

4 - Change the contents of the `DirectoryIndex` directive by adding the following contents to the directive line: `index.php index.phtml`.

Example:

```
<IfModule dir_module>
DirectoryIndex index.html index.php index.phtml
</IfModule>
```

```
#
<IfModule dir_module>
DirectoryIndex index.html index.php index.phtml
</IfModule>
```

5 - Set the `${SRVROOT}` variable with your Apache folder path:

Search for the `Define SRVROOT` and edit it as follows:

```
Define SRVROOT "Apache24"
```

6 - Add the `Options ExecCGI` command below the `Require all granted` line in the `<Directory "${SRVROOT}/htdocs">` directive.

```
Require all granted
Options ExecCGI
</Directory>
```

Example:

```
Options Indexes FollowSymLinks

#
# AllowOverride controls what directives may be placed in the
# .htaccess files. It can be "All", "None", or any combination of the
# AllowOverride FileInfo AuthConfig Limit
#
AllowOverride None

#
# Controls who can get stuff from this server.
#
Require all granted
Options ExecCGI
</Directory>
```

7 - Search for the `<IfModule mime_module>` directive and add the content below **above** this directive:

```
<Directory "/php">
```

```
AllowOverride None
```

```
Options None
Require all granted

</Directory>
```

Example:

```
<Directory "/php">
AllowOverride None
Options None
Require all granted
</Directory>

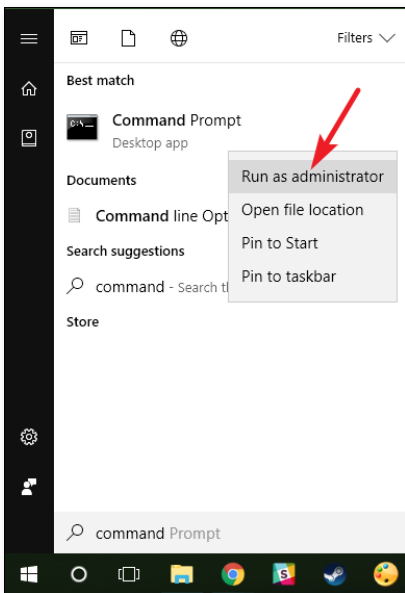
<IfModule mime_module>
```

After performing all these procedures, save the `httpd.conf` file to proceed with Apache installation and startup and PHP configuration.

Install and start Apache

Now that you have set up your apache service, all you have to do is install and start it. Below are the steps that must be followed to successfully complete this process.

1 - Run the Command Prompt (CMD) with Administrator privileges:



2 - Go to the Apache `bin` folder to install the `httpd.exe` executable file:

```
cd C:\Apache24\bin\httpd.exe
```

3 - Install Apache with the command below:

```
httpd.exe -k install
```

Wait for the message **"The 'Apache2.4' service is successfully installed."** to proceed.

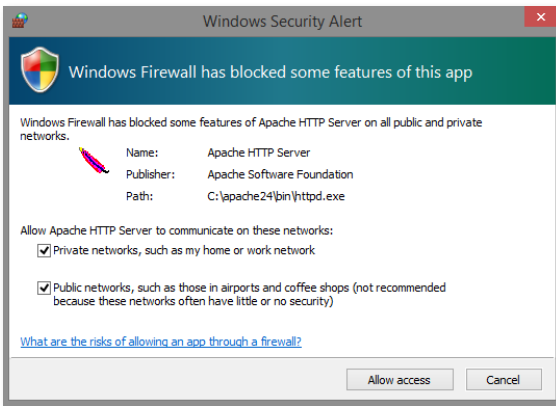
4 - Start the Apache service and verify if it is working.

Use the `httpd.exe -k start` to start your Apache service.

```
C:\Apache24\bin>httpd.exe -k start
C:\Apache24\bin>
```

5 - Grant execution permission to Apache on your Firewall:

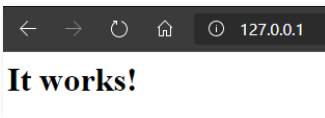
Select the two options available to grant permission, so Apache can function properly.



6 - Check if Apache is active.

To verify that the service is active, simply visit the Apache test page in your browser. You can do it in two ways:

127.0.0.1 or localhost



Configure the PHP 7.3

With Apache active and functional, you now need to define and configure the **PHP.ini** file so those database extensions and libraries are available for use by Scriptcase.

1 - Access the PHP folder in **C:** and rename the **php.ini-development** file to **php.ini**.

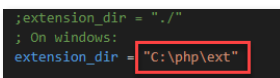


2 - Access the **php.ini** file with a text editor using Administrative privileges and search for the `;extension_dir = "ext"` directive.

Assign the value `"C:\php\ext"` to this directive.

Example:

```
extension_dir = "C:\php\ext"
```



3 - Enable the extensions listed below, by uncommenting them (Remove the semicolon `;` at beginning of line).

Default Extensions

- extension=bz2
- extension=curl
- extension=gd2
- extension=gettext
- extension=imap
- extension=ldap
- extension=fileinfo
- extension=mbstring
- extension=openssl
- extension=exif
- extension=xsl
- extension=soap

Database Extensions:

Extensions	Driver enabled in Scriptcase
<code>extension=mysqli</code>	MySQLi
<code>extension=odbc</code>	DB2 ODBC GENERIC, DB2 ODBC GENERIC 6, Generic ODBC, MS Access ODBC, MS SQL Server ODBC, Oracle ODBC, Progress
<code>extension=pdo_mysql</code>	MySQL PDO
<code>extension=pdo_odbc</code>	DB2 PDO ODBC, Progress PDO ODBC, Sybase PDO ODBC
<code>extension=pdo_pgsql</code>	PostgreSQL PDO
<code>extension=pdo_sqlite</code>	SQLite PDO
<code>extension=pgsql</code>	PostgreSQL 6.3 or below, PostgreSQL 6.4 or above, PostgreSQL 7 ou Acima

NOTE: Some database extensions will depend on third party drivers to be enabled. Click [here](#) to access configuration instructions for your desired database.

4 - Set the recommended minimum value of these PHP directives listed below for Scriptcase to work properly. Search for the directives and assign the value according to this example:

- `max_execution_time = 3600`
- `max_input_time = 3600`
- `max_input_vars = 10000`
- `memory_limit = 1024M`
- `post_max_size = 1024M`
- `upload_max_filesize = 1024M`
- `max_file_uploads = 200`
- `short_open_tag = On`

IMPORTANT: By default PHP can disable some functions that are important for Scriptcase to function properly. Search for the `disable_functions` directive and make sure it is as follows: `disable_functions =` . Click [here](#) to access the list of functions required for Scriptcase to function properly.

5 - Set up PHP **TimeZone** according to your region. You must use the value available in [PHP documentation](#). Search for the `date.timezone` line, uncomment it and edit it according to the selected TimeZone:

Example:

```
date.timezone = America/Recife
```

```
[Date]
; Defines the default timezone used by
; http://php.net/date.timezone
date.timezone = America/Recife
```

6 - Set the folder where temporary files will be stored. Search for the `session.save_path` line, uncomment it, and enter the path to your temporary folder.

Example:

```
session.save_path = "C:\Windows\Temp"
```

```
; http://php.net/session.save-path
session.save_path = "C:\Windows\Temp"
; Whether to use strict session mode.
```

7 - Save all changes made to the `php.ini` file.

8 - Restart the Apache service using the Command Prompt (**CMD**) with Administrator privileges the following command:

```
C:\Apache24\bin\httpd.exe -k restart
```

```
Administrador: Prompt de Comando
Microsoft Windows [versão 10.0.18363.476]
(c) 2019 Microsoft Corporation. Todos os direitos reservados.
C:\WINDOWS\system32>C:\Apache24\bin\httpd.exe -k restart
C:\WINDOWS\system32>
```

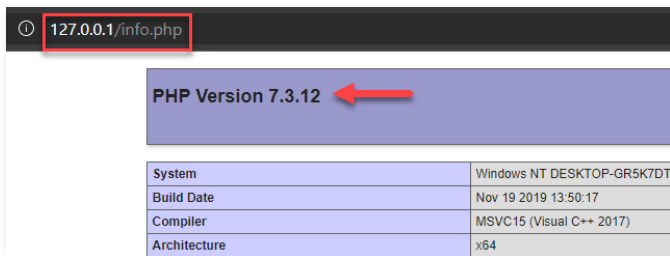
9 - Verify changes made through the `info.php` file. You need to create this file and place it in the `C:\Apache24\htdocs` directory with the following content:

```
<?php
```

```
phpinfo();
```

```
?>
```

After this, check the generated page in your browser by going to the URL `127.0.0.1/info.php` .

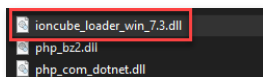


NOTE: For more information about the `phpinfo()` function, check the [documentation](#) do PHP.

Enable the IonCube Loader

Before you start the Scriptcase installation, you need to enable the loader responsible for encrypting Scriptcase in PHP. Follow the steps below to perform this action successfully.

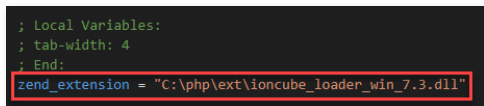
- 1 - Extract the **IonCube Loader** file downloaded at the [beginning](#) of this documentation.
- 2 - Copy the `ioncube_loader_win_7.3.dll` file and paste it into the PHP extensions directory `C:\php\ext`.



- 3 - Edit the `php.ini` file and below the last line enter the path to the extension within the `zend_extension` parameter:

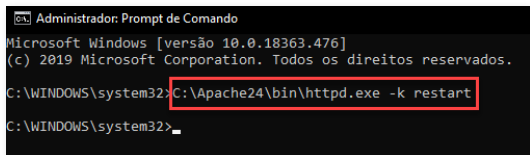
Example:

```
zend_extension = "C:\php\ext\ioncube_loader_win_7.3.dll"
```



- 4 - Restart the Apache service using the Command Prompt (**CMD**) with Administrator privileges the following command:

```
C:\Apache24\bin\httpd.exe -k restart
```



Manual Scriptcase Installation

Listed below are the steps needed to do a Scriptcase manual installation. **To do the steps, it is needed that you have a web server configured in your machine**.

- 1 - Download the Scriptcase (**.zip**) directly from the [download page](#) from our website.
- 2 - Extract the **.zip** file and rename the extracted folder to `scriptcase`.
- 3 - Move the `scriptcase` to your server root. **Depending of your operating system and the installation the path can be different from the example.** However, by default the paths are:

Windows	Server
<code>C:\Apache24\htdocs\</code>	Apache
<code>C:\inetpub\wwwroot\</code>	IIS

Linux	Server
<code>/var/www/html/</code>	Linux Local
<code>/home/\$(whoami)/public_html/</code>	Linux Server

macOS	Server
<code>/Library/WebServer/Documents</code>	Apache

- 4 - Access the Scriptcase using your browser:

```
127.0.0.1/scriptcase
```


- y default the selected language is English.

Installation - Initiating

STEPS	
Initiating	<p>Welcome to ScriptCase!</p> <p>This wizard will guide you in the installation and configuration process of ScriptCase. You will, step by step, be guided on all the stages necessary to adjust its environment to run the tool correctly.</p> <p>Select the language that will be used during this installation</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">English ▼</div>
PHP Extensions	
System Folders	
Database	
ScriptCase Tables	
User	
Finishing	

Next >

- 5 - Check the extensions required for Scriptcase to function and database connection modules that are enabled.

Check if the necessary PHP extensions are loaded.

Required Extensions.	Action.
✓ GD This extension is used to create charts and manipulate images	
✓ MBSTRING This extension is used to convert special chars	
✓ SimpleXML The SimpleXML extension provides a very simple and easily usable toolset to convert XML to an object that can be processed with normal property selectors and array iterators.	
✓ ZIP This extension is used to create and extract files from a ZIP archive	
✓ ZLIB This extension is used to extract files from a ZIP archive	
✓ JSON This extension implements the data interchange format JavaScript Object Notation (JSON).	

Database Extensions.	Action.
✓ com_dotnet Generic ADO, MS Access ADO, MS SQL Server ADO	
✗ ibm_db2 DB2, DB2 ODBC NATIVE	Enable.
✓ interbase Firebird, Interbase 6, Interbase 6.5	
✗ mssql MS SQL Server 7	Enable.
✗ mysql MaxSQL, MySQL (Non-Transactional), MySQL (Transactional)	Enable.
✓ mysqli MySQLi	
✓ oci8 Oracle 8, Oracle 8 Portable, Oracle 8.0.5	
✓ odbc DB2 ODBC GENERIC, DB2 ODBC GENERIC 6, Generic ODBC, MS Access ODBC, MS SQL Server ODBC, Oracle ODBC, Progress	
✗ oracle Oracle 7	Enable.
✓ pdo_dblib Sybase PDO DBLIB	
✓ pdo_firebird Firebird PDO	
✗ pdo_ibm PDO IBM	Enable.
✗ pdo_informix Informix PDO	Enable.

Typical Installation

Check the steps below to complete the manual installation of Scriptcase in a typical way.

NOTE: By selecting this installation, Scriptcase will set your environment by default. Defining the installation database and the login user in Scriptcase.

Typical Installation:
Scriptcase will install everything automatically.

Customized Installation:
You could choose on which database you wish to install ScriptCase and the default username and password to access it.

Next

After clicking proceed, you will be redirected to the Scriptcase Login page.

User

In this field, it is necessary to inform the user to login to Scriptcase. By default, the user is **admin**.

Password

In this field, you must enter the password to login to Scriptcase. By default, the password is **admin**.

Keep me connected

This option keeps the user logged in until they log out through the Scriptcase interface.

Forgot your password?

By clicking this option, it is possible to recover the password of the informed user. An email will be sent to the user in question so they can access the recovery steps.

Important: You can only perform password recovery if you have configured your SMTP information. To access our documentation showing how to perform this configuration click [here](#).

Custom Installation

Check the steps below to complete the manual installation of Scriptcase in a custom manner.

NOTE: By selecting this installation, you will be responsible for setting up your environment. Defining the installation database and the login user for Scriptcase.

Typical Installation:
Scriptcase will install everything automatically.

Customized Installation:
You could choose on which database you wish to install ScriptCase and the default username and password to access it.

[Next](#)

1- Verify that the directories required by Scriptcase have the correct permissions. If any directory listed does not have the active check icon , correct permission for this directory.

Installation :: System Folders

STEPS	
Initiating	ScriptCase performs many operations with files. It is necessary that some directories have the write permission set. Make sure the ScriptCase's directories have the necessary authorization and correct the problem.
PHP Extensions	
System Folders	<input checked="" type="checkbox"/> C:/Apache24/htdocs/scriptcase/app/ <input checked="" type="checkbox"/> C:/Apache24/htdocs/scriptcase/conf/ <input checked="" type="checkbox"/> C:/Apache24/htdocs/scriptcase/backup/ <input checked="" type="checkbox"/> C:/Apache24/htdocs/scriptcase/dev/cont/grp/ <input checked="" type="checkbox"/> C:/Apache24/htdocs/scriptcase/dev/cont/scriptcase/ <input checked="" type="checkbox"/> C:/Apache24/htdocs/scriptcase/dev/cont/ustr/ <input checked="" type="checkbox"/> C:/Apache24/htdocs/scriptcase/file/doc/ <input checked="" type="checkbox"/> C:/Apache24/htdocs/scriptcase/file/img/ <input checked="" type="checkbox"/> C:/Apache24/htdocs/scriptcase/log/ <input checked="" type="checkbox"/> C:/Apache24/htdocs/scriptcase/tmp/
Database	
ScriptCase Tables	
User	
Finishing	

[Next »](#)

Click proceed to proceed with the installation process .

2- Choose the database where the Scriptcase database will be installed. By default, Scriptcase recommends that it be installed with the **SQLite** database.

Installation :: Database

STEPS

- Initiating
- PHP Extensions
- System Folders
- Database**
- ScriptCase Tables
- User
- Finishing

ScriptCase stores its applications data in a database. You must choose the connection parameters for ScriptCase to be able to connect to this database.

DBMS: **SQLite PDO** ✓

We recommend to use SQLite as your main ScriptCase database. Using SQLite, you ensure a greater reliability in ScriptCase features, including **Automatic Backup** of your projects and applications. If you want to install ScriptCase in another database, be aware that the Automatic Backup ScriptCase will not consider your projects and most of the data related to it.

[Change Database](#)

[Next »](#)

- You can change the default database by clicking the **Change Database** button. By clicking this option you will need to set the following options:

Installation :: Database

STEPS

- Initiating
- PHP Extensions
- System Folders
- Database**
- ScriptCase Tables
- User
- Finishing

ScriptCase stores its applications data in a database. You must choose the connection parameters for ScriptCase to be able to connect to this database.

DBMS: **MySQL PDO** ✓

Server: 192.168.254.171:3306

User: root

Password: *****

Database: scriptcase

We recommend to use SQLite as your main ScriptCase database. Using SQLite, you ensure a greater reliability in ScriptCase features, including **Automatic Backup** of your projects and applications. If you want to install ScriptCase in another database, be aware that the Automatic Backup ScriptCase will not consider your projects and most of the data related to it.

[Next »](#)

DBMS

In this option, you will select the Driver to connect to your database.

Server

In this option, you will inform the server to connect to the desired database.

User

You need to inform a user who has access to the database you want Scriptcase to be installed on.

Password

You must enter the authentication password corresponding to the entered user.

Database

You must enter the name of the database where Scriptcase will be installed.

Example:

scriptcase

Clique em prosseguir para avançar no processo de instalação.

3 - Verifique se a criação das tabelas da base de dados do Scriptcase ocorreu com sucesso. As tabelas serão marcadas com o ícone de sucesso ✓ na lateral.

Installation :: ScriptCase Tables

STEPS

- Initiating
- PHP Extensions
- System Folders
- Database
- ScriptCase Tables**
- User
- Finishing

ScriptCase uses tables on a database to store the applications development data.

✓ sc_tbrj	Projects table
✓ sc_tbusu	User Table.
✓ sc_tgroup	Groups table
✓ sc_tuser_group	Relationship table for users and groups
✓ sc_tbatl	User Status Table.
✓ sc_tbapl	Application Table.
✓ sc_tbcnp	Fields Table.
✓ sc_tblog	Table of log schemes
✓ sc_tblog_apl	Applications Backup Table.
✓ sc_tblog_cmp	Backup Table of fields.
✓ sc_tbrep	Data Dictionary Table.
✓ sc_tbrep_tables	Data Dictionary(Tables) Table;
✓ sc_tbrep_fields	Data Dictionary(Fields) Table;
✓ sc_tbconex	Connections Table.
✓ sc_tbeess	Session table
✓ sc_tbvrsao	Project Version Table.
✓ sc_tbevt	Events Table.
✓ sc_tblog_evt	Events Backup Table.
✓ sc_tbttrans	Transactions Table.
✓ sc_tbtodo	Task list table
✓ sc_tbtmsg	Table of messages between users

All ScriptCase tables are already created.

[Next »](#)

Click proceed to proceed with the installation process.

4 - Set the default user who will have access to Scriptcase.

Login

In this option, you will enter the default Scriptcase User Login.

Password

In this option, you must enter the password that will correspond to the user informed.

Confirmation

You must repeat the previously entered password for successful user creation.

Click proceed to proceed with the installation process.

5 - Complete Scriptcase custom installation on your web server.

After clicking proceed, you will be redirected to the Scriptcase Login page.

User

In this field, it is necessary to inform the user to login to Scriptcase. By default, the user is **admin**.

Password

In this field, you must enter the password to login to Scriptcase. By default, the password is **admin**.

Keep me connected

This option keeps the user logged in until they log out through the Scriptcase interface.

Forgot your password?

By clicking this option, it is possible to recover the password of the informed user. An email will be sent to the user in question so they can access the recovery steps.

Important: You can only perform password recovery if you have configured your SMTP information. To access our documentation showing how to perform this configuration click [here](#).

Need to register your Scriptcase?

You can access our [knowledge base](#) to understand all the steps for registering a Scriptcase license.

This documentation will teach all the steps necessary for the installation of the PHP 7.0, Apache 2.4 and the Scriptcase using the manual installation in the Linux environment. **Doing this kind of installation you will be responsible for configuring all your environment settings and the database extensions as well.**

Listed below are the operational systems approved for the Scriptcase usage. **Only these distributions below and their derivatives are approved for the Scriptcase installation.**

Check the list of the supported operating systems:

- edHat\CentOS
- ebian\Ubuntu
- use\OpenSuse

NOTE: In case you need to make a manual installation of the Scriptcase using **PHP 7.0** in Linux, access the documentation in our [knowledge base](#).

Prerequisites

WARNING: If you already have a Web Server with Apache and PHP 7.3 configured, just go to the [lonCube](#) configuration and proceed to manual installation.

To proceed with the installation, you will need to download some files.

Necessaries Files:

- onCube Loader for **Linux (x86 ou x64)**: [Click Here](#)
- criptcase (.zip): [Click Here](#)

NOTE: Before continue with the installation is recommended that you do not have any other web environment (as XAMP, Zend Server, etc) installed in your operational system. **It is recommended to know Linux/Unix environment to use the terminal.**

Configuring the PHP - Ubuntu 19.10\Debian 10

Listed below are the necessaries commands for the manual installation of the PHP 7.3 in the Ubuntu 19.10 or Debian 10. Check correctly the name of the packages in your terminal.

1 - Before run any other command, update your operating system with the following commands:

```
sudo apt-get update
sudo apt-get upgrade
```

2 - Run the command `sudo apt-get install php7.3` to install the PHP 7.3. After that, accept the dependencies installation.

If you are using any Ubuntu version before 19.10, it's necessary to add a repository in your operating system to access the PHP 7.3 packages. See how you can do it below:

```
sudo add-apt-repository ppa:ondrej/php
sudo apt-get update
```

If you are using any Debian version before Debian 10, it's necessary to add a repository in your operating system to access the PHP 7.3 packages. See how you can do it below:

```
sudo apt -y install lsb-release apt-transport-https ca-certificates
sudo wget -O /etc/apt/trusted.gpg.d/php.gpg https://packages.sury.org/php/apt.gpg
echo "deb https://packages.sury.org/php/ $(lsb_release -sc) main" | sudo tee /etc/apt/sources.list.d/php7.3.list
sudo apt-get update
```

```
sudo apt-get install php7.3
```

Example:

```
netnake09@netnake-Vostro:~$ sudo apt-get install php
[sudo] password for netnake09:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  firebird3.0-common firebird3.0-common-doc freetds-common libfbclient2 libbodbci libpq5 libsybdb5 libtonmath1 libzip5
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libapache2-mod-php7.3 php-common php7.3 php7.3-cli php7.3-common php7.3-json php7.3-opcache php7.3-readline
Suggested packages:
  php-pear
The following NEW packages will be installed:
  libapache2-mod-php7.3 php php-common php7.3 php7.3-cli php7.3-common php7.3-json php7.3-opcache php7.3-readline
0 upgraded, 9 newly installed, 0 to remove and 0 not upgraded.
Need to get 0 B/3,923 kB of archives.
After this operation, 17.6 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

3 - Check the PHP version running the command: `php -v`

Example:

```
netmake09@netmake-Vostro:~$ php -v
PHP 7.3.11-0ubuntu0.19.10.1 (cli) (built: Oct 24 2019 11:38:49) ( NTS )
Copyright (c) 1997-2018 The PHP Group
```

4 - Now, install all the extensions below that are required by Scriptcase:

Default Extensions:

- `sudo apt-get install php7.3-curl`
- `sudo apt-get install php7.3-gd`
- `sudo apt-get install php7.3-bcmath`
- `sudo apt-get install php7.3-cgi`
- `sudo apt-get install php7.3-ldap`
- `sudo apt-get install php7.3-mbstring`
- `sudo apt-get install php7.3-xml`
- `sudo apt-get install php7.3-soap`
- `sudo apt-get install php7.3-xsl`
- `sudo apt-get install php7.3-zip`

Database Extensions:

Extension	Driver enabled in Scriptcase
<code>sudo apt-get install php7.3-pgsql</code>	PostgreSQL PDO, PostgreSQL 6.3 or below, PostgreSQL 6.4 or above, PostgreSQL 7 or above
<code>sudo apt-get install php7.3-mysql</code>	MySQLi, MySQL PDO
<code>sudo apt-get install php7.3-sqlite3</code>	SQLite PDO
<code>sudo apt-get install php7.3-interbase</code>	Firebird, Interbase 6, Interbase 6.5, Firebird PDO
<code>sudo apt-get install php7.3-odbc</code>	DB2 ODBC GENERIC, DB2 ODBC GENERIC 6, Generic ODBC, MS Access ODBC, Oracle ODBC, Progress, DB2 PDO ODBC, Progress PDO ODBC, Sybase PDO ODBC
<code>sudo apt-get install php7.3-sybase</code>	DBLIB, Sybase PDO DBLIB

NOTE: Some database extensions will depend on third party drivers to be enabled. Click [here](#) to access the setup instructions for your database.

5 - Access the `php.ini` file in the directory `/etc/php/7.3/apache2/` and set the recommended values for the PHP directives listed below required by the Scriptcase.

Search for these directives and set the values as the example:

- `max_execution_time = 3600`
- `max_input_time = 3600`
- `max_input_vars = 10000`
- `memory_limit = 1024M`
- `post_max_size = 1024M`
- `upload_max_filesize = 1024M`
- `max_file_uploads = 200`
- `short_open_tag = On`

6 - By default, the PHP disable some functions that are required by the Scriptcase. Click [here](#) to access the list of these functions.

- change the line `disable_functions` as the example below:

```
disable_functions =
```

Example:

```
; This directive allows you to disable certain functions for security reasons.
; It receives a comma-delimited list of function names.
; http://php.net/disable-functions
disable_functions =
```

7 - Set up the PHP **Timezone** according your region. Must use the values available in the PHP [documentation](#). Search for the line `date.timezone` in your `php.ini`, uncomment and edit according with the Timezone selected.

Example:

```
date.timezone = America/Recife
```

8 - Set up the folders that will store the temporary files. Search for the line `session.save_path`, uncomment and set the path to your temporary folder.

Example:

```
session.save_path = "/tmp"
```

9 - **Save all the changes made** and restart the Apache service using the following command:

```
sudo service apache2 restart
```

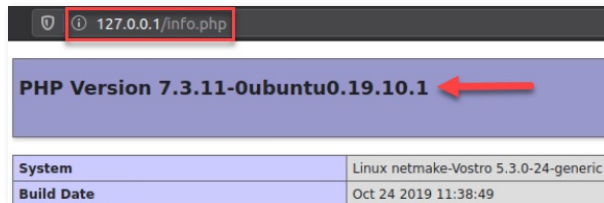
10 - Check if the changes were made using the file `info.php`. You will need to create this file and place it in the `/var/www/html/` path with the following content.

```
<?php
```

```
phpinfo());
```

```
?>
```

After that, check using your browser the page generated accessing the URL `127.0.0.1/info.php`.



NOTE: For more information about the `phpinfo()`, check the [PHP documentation](#).

Setting up the PHP - CentOS 8

Listed below are the necessary commands for the manual installation of the PHP 7.3 in the CentOS 8. Check correctly the name of the packages in your terminal.

1 - Before run any other command, update your operating system with the following commands:

```
sudo yum update
sudo yum upgrade
```

NOTE: In the default CentOS 7/8 repository, the PHP 7.3 is not installed by default. It will need to add the repository to continue the installation.

2 - Add the repository below to download the PHP 7.3 packages and after that update your system:

```
sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
sudo dnf install https://rpms.remirepo.net/enterprise/remi-release-8.rpm
sudo yum update
```

If you use any CentOS version below the 8, it's necessary to add another repository in your operating system to access the PHP 7.3 packages. See how to do it below:

CentOS 6	CentOS 7
<code>sudo yum install epel-release</code>	<code>sudo yum -y install http://rpms.remirepo.net/enterprise/remi-release-7.rpm</code>
<code>sudo rpm -Uvh http://rpms.famillecollet.com/enterprise/remi-release-6.rpm</code>	<code>sudo yum -y install epel-release yum-utils</code>

- In the CentOS 6, it is needed to enable the PHP 7.3 module using the command below:

```
sudo yum --enablerepo=remi-php73 install php
```

3 - Run the commands below to install the **PHP 7.3** and **Apache 2.4** modules and accept their dependencies:

```
sudo dnf module install php:remi-7.3
sudo yum install httpd
```

Example:

```
[root@localhost suporte]# dnf module install php:remi-7.3
Ultima verificação de data de vencimento de metadados: 0:00:29 atrás em qua 11
Dependências resolvidas.
=====
Pacote                Arq.                Versão
-----
Installing group/module packages:
php-cli                x86_64              7.3.12-1.e18.remi
php-common             x86_64              7.3.12-1.e18.remi
php-fpm                x86_64              7.3.12-1.e18.remi
php-json               x86_64              7.3.12-1.e18.remi
php-mbstring           x86_64              7.3.12-1.e18.remi
php-xml                x86_64              7.3.12-1.e18.remi
Instalando dependências:
httpd-filesystem      noarch              2.4.37-12.module_e18.0.0+185+5908
oniguruma              x86_64              6.8.2-1.e18
libargon2              x86_64              20171227-3.e18
Instalando dependências fracas:
nginx-filesystem      noarch              1:1.14.1-9.module_e18.0.0+184+e34
Installing module profiles:
php/common
Enabling module streams:
=====
```

4 - Check the PHP version running the command: `php -v`

Example:

```
[root@localhost html]# php -v
PHP 7.3.12 (cli) (built: Nov 19 2019 10:24:51) ( NTS )
Copyright (c) 1997-2018 The PHP Group
Zend Engine v3.3.12, Copyright (c) 1998-2018 Zend Technologies
[root@localhost html]#
```

5 - Now, install the extensions below required by the Scriptcase:

- o search for packages in versions below CentOS8, you will need to use the command: `yum search php73`

Default Extensions:

- `sudo yum install php-bcmath`
- `sudo yum install php-common`
- `sudo yum install php-cli`
- `sudo yum install php-gd`
- `sudo yum install php-json`
- `sudo yum install php-ldap`
- `sudo yum install php-mbstring`
- `sudo yum install php-soap`
- `sudo yum install php-xml`
- `sudo yum install php-zip`

Database Extensions:

Extension	Driver enabled in the Scriptcase
<code>sudo yum install php-pgsql</code>	PostgreSQL PDO, PostgreSQL 6.3 or below, PostgreSQL 6.4 or above, PostgreSQL 7 or above
<code>sudo yum install php-mysql</code>	MySQLi, MySQL PDO
<code>sudo yum install php-sqlite3</code>	SQLite PDO
<code>sudo yum install php-odbc</code>	DB2 ODBC GENERIC, DB2 ODBC GENERIC 6, Generic ODBC, MS Access ODBC, Oracle ODBC, Progress, DB2 PDO ODBC, Progress PDO ODBC, Sybase PDO ODBC
<code>sudo yum install php-pdo-dblib</code>	DBLIB, Sybase PDO DBLIB

NOTE: Some database extensions will depend on third party drivers to be enabled. Click [here](#) to access the setup instructions for your database.

6 - Access the `php.ini` file in the directory `/etc/` and set the recommended values for the PHP directives listed below required by the Scriptcase.

Search for these directives and set the values as the example:

- `max_execution_time = 3600`
- `max_input_time = 3600`
- `max_input_vars = 10000`
- `memory_limit = 1024M`
- `post_max_size = 1024M`
- `upload_max_filesize = 1024M`
- `max_file_uploads = 200`
- `short_open_tag = On`

7 - By default, the PHP disable some functions that are required by the Scriptcase. Click [here](#) to access the list of these functions.

- change the line `disable_functions` as the example below:

```
disable_functions =
```

Example:

```
; This directive allows you to disable certain functions for security reasons.
; It receives a comma-delimited list of function names.
; http://php.net/disable-functions
disable_functions =
```

8 - Set up the PHP `TimeZone` according your region. Must use the values available in the PHP [documentation](#). Search for the line `date.timezone` in your `php.ini`, uncomment and edit according with the TimeZone selected.

Example:

```
date.timezone = America/Recife
```

8 - Set up the folders that will store the temporary files. Search for the line `session.save_path`, uncomment and set the path to your temporary folder.

Example:

```
session.save_path = "/tmp"
```

9 - Save all the changes made and restart the Apache service using the following command:

```
sudo systemctl restart httpd
```

10 - Check if the changes were made using the file `info.php`. You will need to create this file and place it in the `/var/www/html/` path with the following content.

```
<?php
phpinfo();
?>
```

After that, check using your browser the page generated accessing the URL `127.0.0.1/info.php`.



NOTE: For more information about the `phpinfo()`, check the [PHP documentation](#).

Enabling the IonCube Loader

Before starting the Scriptcase installation, you need to enable the loader used for the Scriptcase encryption in your PHP. Follow the steps below showing how to do it.

1 - Access the Downloads folder and extract the file downloaded referent to the **IonCube Loader** at the [beginning](#) of this documentation.

Example:

```
cd /home/User/Downloads
```

```
sudo chmod 777 ioncube_loaders_lin_x86-64.zip && unzip ioncube_loaders_lin_x86-64.zip
```

2 - Copy the file `ioncube_loader_lin_7.3.so` and past in the `extension_dir` directory of your PHP. We will use the path `/usr/lib/php/20180731` for this example.

Example:

Ubuntu 19.10\Debian 8	CentOS 7/8
<code>/usr/lib/php/20180731</code>	<code>/usr/lib64/php/modules</code>

```
sudo cp ioncube_loader_lin_7.3.so /usr/lib/php/20180731
```

3 - Edit the `php.ini` file and below the last line set the path to the extension in the `zend_extension` parameter:

Example:

Ubuntu 19.10\Debian 8	CentOS 7/8
<code>/etc/php/7.3/apache2/</code>	<code>/etc/</code>

```
sudo nano /etc/php/7.3/apache2/php.ini
```

```
zend_extension = "/usr/lib/php/20180731/ioncube_loader_lin_7.3.so"
```

4 - Restart the Apache service using the following command:

Ubuntu 19.10\Debian 8	CentOS 7/8
<code>sudo service apache2 restart</code>	<code>sudo systemctl restart httpd</code>

IMPORTANT: To enable the IonCube in the CentOS, the `SELinux` service must be **disabled**. Check the service status using the command: `sestatus`.

To disabled the service, access the file settings using the command `sudo nano /etc/selinux/config` and change the `SELINUX` and `SELINUXTYPE` directives as this: `SELINUX=disabled` e `SELINUXTYPE=minimum`. After that, save the file and restart your Centos to apply the changes and enable the IonCube.

Scriptcase Manual Installation

Listed below are the steps needed to do a Scriptcase manual installation. **To do the steps, it is needed that you have a web server configured in your machine**.

1 - Download the Scriptcase (**zip**) directly from the [download page](#) from our website.

2 - Extract the **.zip** file and rename the extracted folder to `scriptcase`.

3 - Move the `scriptcase` to your server root. **Depending of your operating system and the installation the path can be different from the example.** However, by default the paths are:

Windows	Server
<code>C:\Apache24\htdocs\</code>	Apache
<code>C:\inetpub\wwwroot\</code>	IIS

Linux	Server
<code>/var/www/html/</code>	Linux Local
<code>/home/\${whoami}/public_html/</code>	Linux Server

macOS	Server
<code>/Library/WebServer/Documents</code>	Apache

4 - Access the Scriptcase using your browser:

`127.0.0.1/scriptcase`

- y default the selected language is English.

Installation - Initiating

STEPS	
Initiating	<p>Welcome to ScriptCase!</p> <p>This wizard will guide you in the installation and configuration process of ScriptCase. You will, step by step, be guided on all the stages necessary to adjust its environment to run the tool correctly.</p> <p>Select the language that will be used during this installation</p> <p>English ▼</p> <p>Next »</p>
PHP Extensions	
System Folders	
Database	
ScriptCase Tables	
User	
Finishing	

5 - Check the extensions required for Scriptcase to function and database connection modules that are enabled.

Check if the necessary PHP extensions are loaded.

Required Extensions.	Action.
<input checked="" type="checkbox"/> GD This extension is used to create charts and manipulate images	
<input checked="" type="checkbox"/> MBSTRING This extension is used to convert special chars	
<input checked="" type="checkbox"/> SimpleXML The SimpleXML extension provides a very simple and easily usable toolset to convert XML to an object that can be processed with normal property selectors and array iterators.	
<input checked="" type="checkbox"/> ZIP This extension is used to create and extract files from a ZIP archive	
<input checked="" type="checkbox"/> ZLIB This extension is used to extract files from a ZIP archive	
<input checked="" type="checkbox"/> JSON This extension implements the data interchange format JavaScript Object Notation (JSON).	

Database Extensions.	Action.
<input checked="" type="checkbox"/> com_dotnet Generic ADO, MS Access ADO, MS SQL Server ADO	
<input checked="" type="checkbox"/> ibm_db2 DB2, DB2 ODBC NATIVE	Enable.
<input checked="" type="checkbox"/> interbase Firebird, Interbase 6, Interbase 6.5	
<input checked="" type="checkbox"/> mssql MS SQL Server 7	Enable.
<input checked="" type="checkbox"/> mysql MaxSQL, MySQL (Non-Transactional), MySQL (Transactional)	Enable.
<input checked="" type="checkbox"/> mysqli MySQLi	
<input checked="" type="checkbox"/> oci8 Oracle 8, Oracle 8 Portable, Oracle 8.0.5	
<input checked="" type="checkbox"/> odbc DB2 ODBC GENERIC, DB2 ODBC GENERIC 6, Generic ODBC, MS Access ODBC, MS SQL Server ODBC, Oracle ODBC, Progress	
<input checked="" type="checkbox"/> oracle Oracle 7	Enable.
<input checked="" type="checkbox"/> pdo_dblib Sybase PDO DBLIB	
<input checked="" type="checkbox"/> pdo_firebird Firebird PDO	
<input checked="" type="checkbox"/> pdo_ibm PDO IBM	Enable.
<input checked="" type="checkbox"/> pdo_informix Informix PDO	Enable.

Typical Installation

Check the steps below to complete the manual installation of Scriptcase in a typical way.

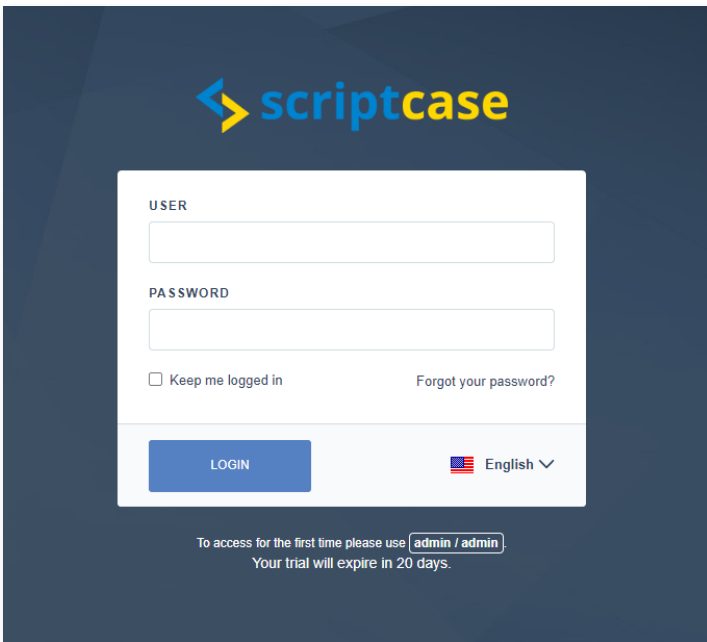
NOTE: By selecting this installation, Scriptcase will set your environment by default. Defining the installation database and the login user in Scriptcase.

Typical Installation:
Scriptcase will install everything automatically.

Customized Installation:
You could choose on which database you wish to install ScriptCase and the default username and password to access it.

[Next](#)

After clicking proceed, you will be redirected to the Scriptcase Login page.



User

In this field, it is necessary to inform the user to login to Scriptcase. By default, the user is **admin**.

Password

In this field, you must enter the password to login to Scriptcase. By default, the password is **admin**.

Keep me connected

This option keeps the user logged in until they log out through the Scriptcase interface.

Forgot your password?

By clicking this option, it is possible to recover the password of the informed user. An email will be sent to the user in question so they can access the recovery steps.

Important: You can only perform password recovery if you have configured your SMTP information. To access our documentation showing how to perform this configuration click [here](#).

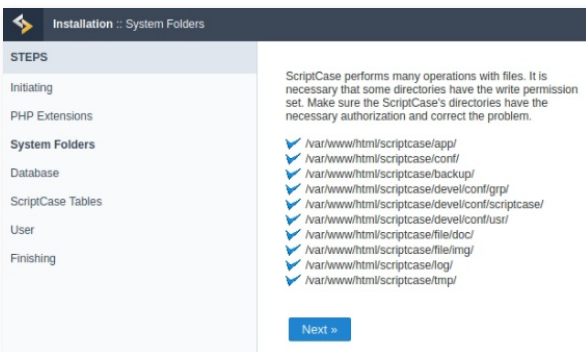
Customized Installation

Check the steps below to complete the manual installation of Scriptcase in a custom manner.

NOTE: By selecting this installation, you will be responsible for setting up your environment. Defining the installation database and the login user for Scriptcase.

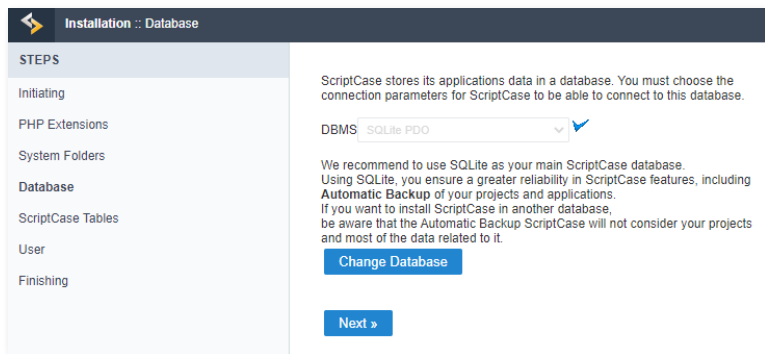


1- Verify that the directories required by Scriptcase have the correct permissions. If any directory listed does not have the active check icon , correct permission for this directory.

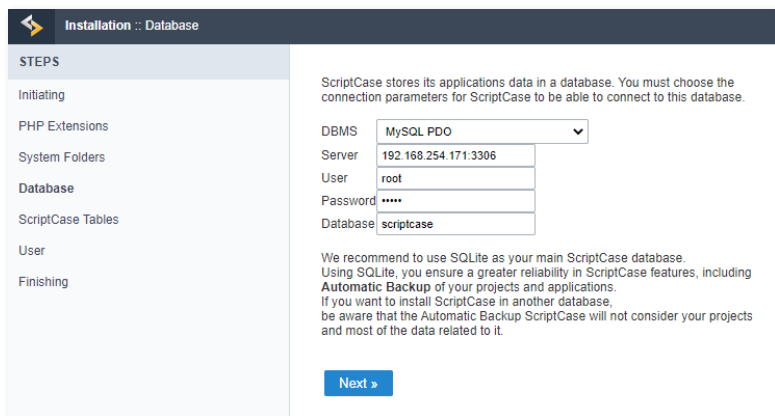


Click proceed to proceed with the installation process .

2- Choose the database where the Scriptcase database will be installed. By default, Scriptcase recommends that it be installed with the **SQLite** database.



- You can change the default database by clicking the **Change Database** button. By clicking this option you will need to set the following options:



DBMS

In this option, you will select the Driver to connect to your database.

Server

In this option, you will inform the server to connect to the desired database.

User

You need to inform a user who has access to the database you want Scriptcase to be installed on.

Password

You must enter the authentication password corresponding to the entered user.

Database

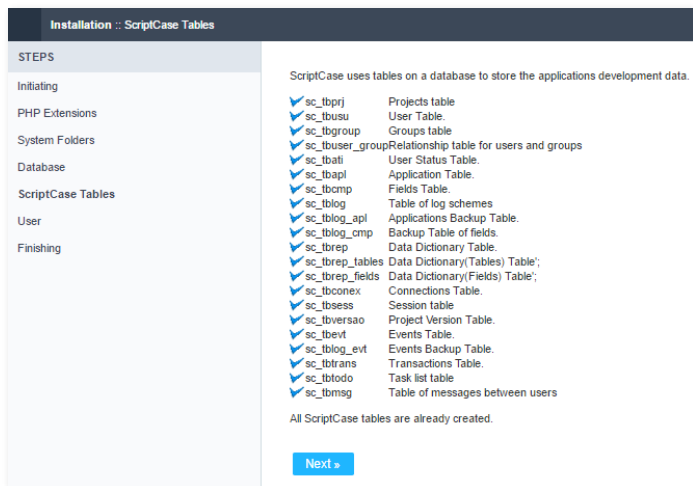
You must enter the name of the database where Scriptcase will be installed.

Example:

`scriptcase`

Clique em prosseguir para avançar no processo de instalação.

3 - Verifique se a criação das tabelas da base de dados do Scriptcase ocorreu com sucesso. As tabelas serão marcadas com o ícone de sucesso ✓ na lateral.



Click proceed to proceed with the installation process.

4 - Set the default user who will have access to Scriptcase.

Login

In this option, you will enter the default Scriptcase User Login.

Password

In this option, you must enter the password that will correspond to the user informed.

Confirmation

You must repeat the previously entered password for successful user creation.

Click proceed to proceed with the installation process.

5 - Complete Scriptcase custom installation on your web server.

After clicking proceed, you will be redirected to the Scriptcase Login page.

User

In this field, it is necessary to inform the user to login to Scriptcase. By default, the user is **admin**.

Password

In this field, you must enter the password to login to Scriptcase. By default, the password is **admin**.

Keep me connected

This option keeps the user logged in until they log out through the Scriptcase interface.

Forgot your password?

By clicking this option, it is possible to recover the password of the informed user. An email will be sent to the user in question so they can access the recovery steps.

Important: You can only perform password recovery if you have configured your SMTP information. To access our documentation showing how to perform this configuration click [here](#).

Do you need to register your Scriptcase?

You can access our [knowledge base](#) to understand all the steps to register the Scriptcase serial.

This documentation will cover how installing PHP 7.3 and ScriptCase manually on macOS. **Performing this type of installation is your responsible for configuring your entire environment as well as database extensions..**

Listed below are versions of macOS approved for use by Scriptcase.

Our supported versions:

- macOS v10.13 "High Sierra"
- macOS v10.14 "Mojave"
- macOS 10.15 "Catalina"

NOTE: If you want access our documentation about manual install of ScriptCase **PHP 7.0** in Linux, see through our [knowledge base](#).

Requirements

WARNING: If you already have a Web Server with Apache and PHP 7.3 configured, just go to the [lonCube](#) configuration and proceed to manual installation.

To proceed with the installation, you need to download some files.

Required Files:

- onCube Loader for **OS X (x86 ou x64)**: [Click here](#)
- criptcase (.zip): [Click here](#)

NOTE: Before proceeding with the installation it is recommended not to have any other web environments (such as XAMP, Zend Server, etc.) installed on the operating system. **Knowledge of Unix environment is recommended to use terminal.**

PHP configuration

Listed below are the commands required for the manual installation of PHP 7.3 on macOS. Check the package naming on your macOS terminal correctly.

1 - You'll need to install **Homebrew** to be able to access PHP 7.3 and other packages. Install **Brew** with the following commands in the macOS terminal:

```
/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

2 - In the macOS terminal, execute `brew update` to update **Brew**.

3 - Use the command `brew install php@7.3` for installing PHP 7.3.

4 - After the installation process completes, execute `brew info php@7.3` to get the information that we need to proceed with environment setting up.

```
netmake@Mac-mini-de-Netmake ~ % brew info php@7.3
php@7.3: stable 7.3.12 (bottled) [keg-only]
General-purpose scripting language
https://www.php.net/
/usr/local/Cellar/php@7.3/7.3.12 (521 files, 77MB)
  Poured from bottle on 2019-12-11 at 20:11:30
From: https://github.com/Homebrew/homebrew-core/blob/master/Formula/php@7.3.rb
=> Dependencies
Build: httpd ✗, pkg-config ✗
Required: apr ✓, apr-util ✓, argon2 ✓, aspell ✓, autoconf ✓, curl-openssl ✓, freetds
libzip ✓, openldap ✓, openssl@1.1 ✓, sqlite ✓, tidy-html5 ✓, unixodbc ✓, webp ✓
=> Caveats
To enable PHP in Apache add the following to httpd.conf and restart Apache:
  LoadModule php7_module /usr/local/opt/php@7.3/lib/httpd/modules/libphp7.so
```

5 - You need to configure Apache so that PHP 7.3 that was installed in the previous steps will be used as default on your system.

- edit the Apache file `httpd.conf` and comment out the path to the old PHP 7.3 module if present in the file. To comment, you need to add a `#` at the beginning of the line.

Example:

```
sudo nano /etc/apache2/httpd.conf
```

```
#LoadModule php7_module libexec/apache2/libphp7.so
```

6 - Add path to PHP 7.3 module installed after last line `LoadModule` in `httpd.conf`.

Example:

```
LoadModule php7_module /usr/local/opt/php@7.3/lib/httpd/modules/libphp7.so
```



```
LoadModule alias_module libexec/apache2/mod_alias.so
#LoadModule rewrite_module libexec/apache2/mod_rewrite.so
#LoadModule php7_module libexec/apache2/libphp7.so
#LoadModule perl_module libexec/apache2/mod_perl.so
LoadModule hfs_apple_module libexec/apache2/mod_hfs_apple.so
LoadModule php7_module /usr/local/opt/php@7.3/lib/httpd/modules/libphp7.so

<IfModule unixd_module>
```

After performing this action, save the file and restart the apache service..

```
sudo apachectl restart
```

7 - Add the path to the installation of PHP 7.3 in the `~/.zshrc` file with the following commands::

```
sudo nano ~/.zshrc
```

```
echo 'export PATH="/usr/local/opt/php@7.3/bin:$PATH"' >> ~/.zshrc echo 'export PATH="/usr/local/opt/php@7.3/sbin:$PATH"' >> ~/.zshrc
```

8 - Edit the file `~/.bash_profile` and add PHP 7.3 path with these following commands:

```
sudo nano ~/.bash_profile
```

```
export PATH="/usr/local/opt/php@7.3/bin:$PATH"
```

9 - Verify PHP version by running command: `php -v`

Example:

```
netmake@Mac-mini-de-Netmake ~ % php -v
PHP 7.3.12 (cli) (built: Nov 29 2019 20:07:49) ( NTS )
Copyright (c) 1997-2018 The PHP Group
Zend Engine v3.3.12, Copyright (c) 1998-2018 Zend Technologies
with Zend OPcache v7.3.12, Copyright (c) 1999-2018, by Zend Technologies
```

10 - Access the `php.ini` file in the directory. `/usr/local/etc/php/7.3/` and set the recommended minimum value correctly of these PHP directives listed below for Scriptcase work properly.

Search for the directives and assign the value according to this example:

- `max_execution_time = 3600`
- `max_input_time = 3600`
- `max_input_vars = 10000`
- `memory_limit = 1024M`
- `post_max_size = 1024M`
- `upload_max_filesize = 1024M`
- `max_file_uploads = 200`
- `short_open_tag = On`

11 - By default, PHP disables some functions that are important to Scriptcase. Click [here](#) to access the necessary functions list for ScriptCase

- change item line `disable_functions` as example below:

```
disable_functions =
```

Example:

```
; This directive allows you to disable certain functions for security reasons.
; It receives a comma-delimited list of function names.
; http://php.net/disable-functions
disable_functions =
```

12 - Set PHP `TimeZone` according to your region. Use the value available in the PHP [documentation](#). Search for `date.timezone` in your `php.ini` file, uncomment it and edit according to the selected TimeZone.

Example:

```
date.timezone = America/Recife
```

13 - Set the folder where temporary files will be stored. search for `session.save_path`, uncoment it and inform your temporary folder path.

Example:

```
session.save_path = "/tmp"
```

14 - Save all modifications made and restart the Apache service using the following command in terminal:

```
sudo apachectl restart
```

15 - Verify changes made through the file `info.php`. You need to create this file and put it in the directory. `/Library/WebServer/Documents/` with the following content:

```
<?php
```

```
phpinfo();
```

```
?>
```

After that, check in your browser the page generated by accessing the URL `127.0.0.1/info.php`.



NOTE: For more information about the function `phpinfo()`, verify our PHP [documentation](#).

Enable IonCube Loader

Before you start installing Scriptcase, you need to enable the loader responsible for encrypting Scriptcase in PHP. Follow the steps below to perform this action successfully.

1 - Go to the Downloads folder and extract the downloaded file for **IonCube Loader** at the beginning of this documentation [start](#).

Example:

```
cd /Users/usuario/Downloads
```

```
sudo chmod 777 ioncube_loaders_dar_x86-64.zip && unzip ioncube_loaders_dar_x86-64.zip
```

```
cd ioncube
```

2 - Copy the file `ioncube_loader_dar_7.3.so` and paste it into your PHP `extension_dir` directory. We'll use `/usr/local/lib/php/pecl/20180731` for this example.

Example:

```
sudo cp ioncube_loader_dar_7.3.so /usr/local/lib/php/pecl/20180731
```

3 - Edit the `php.ini` file and below the last line enter the path to the extension within the parameter `zend_extension`:

Example:

```
sudo nano /usr/local/etc/php/7.3/php.ini
```

```
zend_extension = "/usr/local/lib/php/pecl/20180731/ioncube_loader_dar_7.3.so"
```

4 - Restart the Apache service with the following terminal command:

```
sudo apachectl restart
```

Manual ScriptCase installation

Listed below are the steps needed to do a Scriptcase manual installation. **To do the steps, it is needed that you have a web server configured in your machine**.

1 - Download the Scriptcase (**zip**) directly from the [download page](#) from our website.

2 - Extract the **.zip** file and rename the extracted folder to `scriptcase`.

3 - Move the `scriptcase` to your server root. **Depending of your operating system and the installation the path can be different from the example.** However, by default the paths are:

Windows	Server
<code>C:\Apache24\htdocs\</code>	Apache
<code>C:\inetpub\wwwroot\</code>	IIS

Linux	Server
<code>/var/www/html/</code>	Linux Local
<code>/home/\${whoami}/public_html/</code>	Linux Server

macOS	Server
<code>/Library/WebServer/Documents</code>	Apache

4 - Access the Scriptcase using your browser:

```
127.0.0.1/scriptcase
```

- **y** default the selected language is English.

Installation - Initiating

STEPS

Initiating

Welcome to ScriptCase!

This wizard will guide you in the installation and configuration process of ScriptCase. You will, step by step, be guided on all the stages necessary to adjust its environment to run the tool correctly.

Select the language that will be used during this installation

English ▼

[Next >](#)

PHP Extensions

System Folders

Database

ScriptCase Tables

User

Finishing

5 - Check the extensions required for Scriptcase to function and database connection modules that are enabled.

Check if the necessary PHP extensions are loaded.

Required Extensions.	Action.
✓ GD This extension is used to create charts and manipulate images	
✓ MBSTRING This extension is used to convert special chars	
✓ SimpleXML The SimpleXML extension provides a very simple and easily usable toolset to convert XML to an object that can be processed with normal property selectors and array iterators.	
✓ ZIP This extension is used to create and extract files from a ZIP archive	
✓ ZLIB This extension is used to extract files from a ZIP archive	
✓ JSON This extension implements the data interchange format JavaScript Object Notation (JSON).	

Database Extensions.	Action.
✓ com_dotnet Generic ADO, MS Access ADO, MS SQL Server ADO	
✗ ibm_db2 DB2, DB2 ODBC NATIVE	Enable.
✓ interbase Firebird, Interbase 6, Interbase 6.5	
✗ mssql MS SQL Server 7	Enable.
✗ mysql MaxSQL, MySQL (Non-Transactional), MySQL (Transactional)	Enable.
✓ mysqli MySQLi	
✓ oci8 Oracle 8, Oracle 8 Portable, Oracle 8.0.5	
✓ odbc DB2 ODBC GENERIC, DB2 ODBC GENERIC 6, Generic ODBC, MS Access ODBC, MS SQL Server ODBC, Oracle ODBC, Progress	
✗ oracle Oracle 7	Enable.
✓ pdo_dblib Sybase PDO DBLIB	
✓ pdo_firebird Firebird PDO	
✗ pdo_ibm PDO IBM	Enable.
✗ pdo_informix Informix PDO	Enable.

Typical installation

Check the steps below to complete the manual installation of Scriptcase in a typical way.

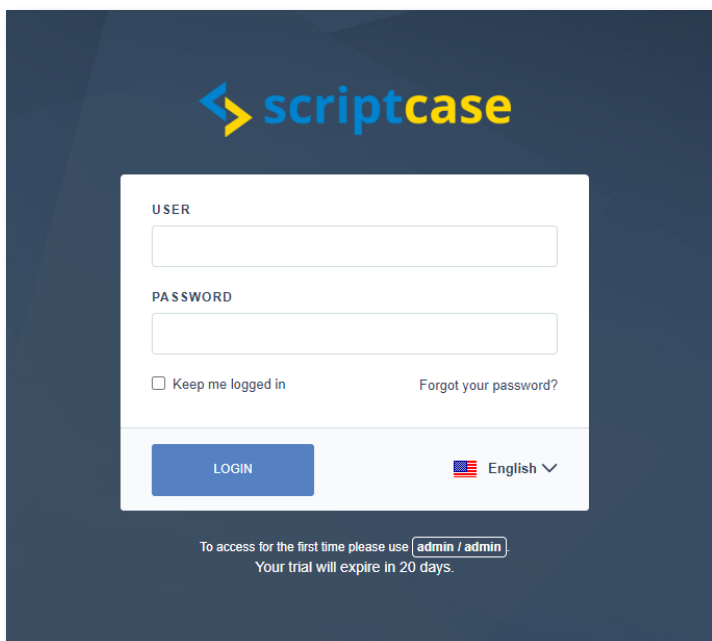
NOTE: By selecting this installation, Scriptcase will set your environment by default. Defining the installation database and the login user in Scriptcase.

Typical Installation:
Scriptcase will install everything automatically.

Customized Installation:
You could choose on which database you wish to install ScriptCase and the default username and password to access it.

[Next](#)

After clicking proceed, you will be redirected to the Scriptcase Login page.



The image shows the Scriptcase login interface. At the top is the Scriptcase logo. Below it is a white login form with the following elements:

- USER**: A text input field.
- PASSWORD**: A text input field.
- Keep me logged in
- [Forgot your password?](#)
- LOGIN**: A blue button.
- Language selector: English ▾

Below the form, a message states: "To access for the first time please use `admin / admin`. Your trial will expire in 20 days."

User

In this field, it is necessary to inform the user to login to Scriptcase. By default, the user is **admin**.

Password

In this field, you must enter the password to login to Scriptcase. By default, the password is **admin**.

Keep me connected

This option keeps the user logged in until they log out through the Scriptcase interface.

Forgot your password?

By clicking this option, it is possible to recover the password of the informed user. An email will be sent to the user in question so they can access the recovery steps.

Important: You can only perform password recovery if you have configured your SMTP information. To access our documentation showing how to perform this configuration click [here](#).

Custom installation

Check the steps below to complete the manual installation of Scriptcase in a custom manner.

NOTE: By selecting this installation, you will be responsible for setting up your environment. Defining the installation database and the login user for Scriptcase.

Typical Installation:
Scriptcase will install everything automatically.

Customized Installation:
You could choose on which database you wish to install ScriptCase and the default username and password to access it.

Next

1- Verify that the directories required by Scriptcase have the correct permissions. If any directory listed does not have the active check icon , correct permission for this directory.

Instalação :: Diretórios do Sistema

PASSOS

- Inicialização
- Extensões do PHP
- Diretórios do Sistema**
- Banco de Dados
- Tabelas do ScriptCase
- Usuário
- Finalização

O ScriptCase realiza diversas operações em arquivo. Para isto, é necessário que alguns diretórios tenham a permissão de escrita liberada.

Verifique se algum dos diretórios usados pelo ScriptCase não possui a autorização necessária e corrija o problema.

- /Library/WebServer/Documents/scriptcase9/app/
- /Library/WebServer/Documents/scriptcase9/conf/
- /Library/WebServer/Documents/scriptcase9/backup/
- /Library/WebServer/Documents/scriptcase9/devell/conf/grp/
- /Library/WebServer/Documents/scriptcase9/devell/conf/scriptcase/
- /Library/WebServer/Documents/scriptcase9/devell/conf/user/
- /Library/WebServer/Documents/scriptcase9/file/doc/
- /Library/WebServer/Documents/scriptcase9/file/img/
- /Library/WebServer/Documents/scriptcase9/log/
- /Library/WebServer/Documents/scriptcase9/tmp/

Prosseguir >

Clique em prosseguir para avançar no processo de instalação.

2- Choose the database where the Scriptcase database will be installed. By default, Scriptcase recommends that it be installed with the **SQLite** database.

Installation :: Database

STEPS

- Initiating
- PHP Extensions
- System Folders
- Database**
- ScriptCase Tables
- User
- Finishing

ScriptCase stores its applications data in a database. You must choose the connection parameters for ScriptCase to be able to connect to this database.

DBMS: **SQLite PDO** ✓

We recommend to use SQLite as your main ScriptCase database. Using SQLite, you ensure a greater reliability in ScriptCase features, including **Automatic Backup** of your projects and applications. If you want to install ScriptCase in another database, be aware that the Automatic Backup ScriptCase will not consider your projects and most of the data related to it.

[Change Database](#)

[Next »](#)

- You can change the default database by clicking the **Change Database** button. By clicking this option you will need to set the following options:

Installation :: Database

STEPS

- Initiating
- PHP Extensions
- System Folders
- Database**
- ScriptCase Tables
- User
- Finishing

ScriptCase stores its applications data in a database. You must choose the connection parameters for ScriptCase to be able to connect to this database.

DBMS: **MySQL PDO** ▼

Server: **192.168.254.171:3306**

User: **root**

Password: *********

Database: **scriptcase**

We recommend to use SQLite as your main ScriptCase database. Using SQLite, you ensure a greater reliability in ScriptCase features, including **Automatic Backup** of your projects and applications. If you want to install ScriptCase in another database, be aware that the Automatic Backup ScriptCase will not consider your projects and most of the data related to it.

[Next »](#)

DBMS

In this option, you will select the Driver to connect to your database.

Server

In this option, you will inform the server to connect to the desired database.

User

You need to inform a user who has access to the database you want Scriptcase to be installed on.

Password

You must enter the authentication password corresponding to the entered user.

Database

You must enter the name of the database where Scriptcase will be installed.

Example:

`scriptcase`

Clique em prosseguir para avançar no processo de instalação.

3 - Verifique se a criação das tabelas da base de dados do Scriptcase ocorreu com sucesso. As tabelas serão marcadas com o ícone de sucesso  na lateral.

Installation :: ScriptCase Tables

STEPS

- Initiating
- PHP Extensions
- System Folders
- Database
- ScriptCase Tables**
- User
- Finishing

ScriptCase uses tables on a database to store the applications development data.

- ✓ sc_tbrj Projects table
- ✓ sc_tbusu User Table.
- ✓ sc_tbrgroup Groups table
- ✓ sc_tbruser_group Relationship table for users and groups
- ✓ sc_tbrati User Status Table.
- ✓ sc_tbrapi Application Table.
- ✓ sc_tbrcmp Fields Table.
- ✓ sc_tbrlog Table of log schemes
- ✓ sc_tbrlog_apl Applications Backup Table.
- ✓ sc_tbrlog_cmp Backup Table of fields.
- ✓ sc_tbrrep Data Dictionary Table.
- ✓ sc_tbrrep_tables Data Dictionary (Tables) Table;
- ✓ sc_tbrrep_fields Data Dictionary (Fields) Table;
- ✓ sc_tbrconex Connections Table.
- ✓ sc_tbrsess Session table
- ✓ sc_tbrversao Project Version Table.
- ✓ sc_tbrbev Events Table.
- ✓ sc_tbrlog_evt Events Backup Table.
- ✓ sc_tbrtrans Transactions Table.
- ✓ sc_tbrtodo Task list table
- ✓ sc_tbrmsg Table of messages between users

All ScriptCase tables are already created.

[Next »](#)

Click proceed to proceed with the installation process.

_4 - Set the default user who will have access to Scriptcase.

Login

In this option, you will enter the default Scriptcase User Login.

Password

In this option, you must enter the password that will correspond to the user informed.

Confirmation

You must repeat the previously entered password for successful user creation.

Click proceed to proceed with the installation process.

5 - Complete Scriptcase custom installation on your web server.

After clicking proceed, you will be redirected to the Scriptcase Login page.

User

In this field, it is necessary to inform the user to login to Scriptcase. By default, the user is **admin**.

Password

In this field, you must enter the password to login to Scriptcase. By default, the password is **admin**.

Keep me connected

This option keeps the user logged in until they log out through the Scriptcase interface.

Forgot your password?

By clicking this option, it is possible to recover the password of the informed user. An email will be sent to the user in question so they can access the recovery steps.

Important: You can only perform password recovery if you have configured your SMTP information. To access our documentation showing how to perform this configuration click [here](#).

Needing register your ScriptCase?

You can access our [knowledge base](#) to understand all the steps for registering Scriptcase license.

Listed below are the steps needed to do a Scriptcase manual installation. **To do the steps, it is needed that you have a web server configured in your machine .**

- 1 - Download the Scriptcase (**zip**) directly from the [download page](#) from our website.
- 2 - Extract the **.zip** file and rename the extracted folder to **scriptcase** .
- 3 - Move the **scriptcase** to your server root. **Depending of your operating system and the installation the path can be different from the example.** However, by default the paths are:

Windows	Server
C:\Apache24\htdocs\	Apache
C:\inetpub\wwwroot\	IIS

Linux	Server
/var/www/html/	Linux Local
/home/\${whoami}/public_html/	Linux Server

macOS	Server
/Library/WebServer/Documents	Apache

- 4 - Access the Scriptcase using your browser:

127.0.0.1/scriptcase

- **y default the selected language is English.**

Installation - Initiating

STEPS	
Initiating	<p>Welcome to ScriptCase!</p> <p>This wizard will guide you in the installation and configuration process of ScriptCase. You will, step by step, be guided on all the stages necessary to adjust its environment to run the tool correctly.</p> <p>Select the language that will be used during this installation</p> <p>English</p> <p>Next ></p>
PHP Extensions	
System Folders	
Database	
ScriptCase Tables	
User	
Finishing	

- 5 - Check the extensions required for Scriptcase to function and database connection modules that are enabled.

Check if the necessary PHP extensions are loaded.

Required Extensions.	Action.
<input checked="" type="checkbox"/> GD This extension is used to create charts and manipulate images	
<input checked="" type="checkbox"/> MBSTRING This extension is used to convert special chars	
<input checked="" type="checkbox"/> SimpleXML The SimpleXML extension provides a very simple and easily usable toolset to convert XML to an object that can be processed with normal property selectors and array iterators.	
<input checked="" type="checkbox"/> ZIP This extension is used to create and extract files from a ZIP archive	
<input checked="" type="checkbox"/> ZLIB This extension is used to extract files from a ZIP archive	
<input checked="" type="checkbox"/> JSON This extension implements the data interchange format JavaScript Object Notation (JSON).	

Database Extensions.	Action.
<input checked="" type="checkbox"/> com_dotnet Generic ADO, MS Access ADO, MS SQL Server ADO	
<input checked="" type="checkbox"/> ibm_db2 DB2, DB2 ODBC NATIVE	Enable.
<input checked="" type="checkbox"/> interbase Firebird, Interbase 6, Interbase 6.5	
<input checked="" type="checkbox"/> mssql MS SQL Server 7	Enable.
<input checked="" type="checkbox"/> mysql MaxSQL, MySQL (Non-Transactional), MySQL (Transactional)	Enable.
<input checked="" type="checkbox"/> mysqli MySQLi	
<input checked="" type="checkbox"/> oci8 Oracle 8, Oracle 8 Portable, Oracle 8.0.5	
<input checked="" type="checkbox"/> odbc DB2 ODBC GENERIC, DB2 ODBC GENERIC 6, Generic ODBC, MS Access ODBC, MS SQL Server ODBC, Oracle ODBC, Progress	
<input checked="" type="checkbox"/> oracle Oracle 7	Enable.
<input checked="" type="checkbox"/> pdo_dblib Sybase PDO DBLIB	
<input checked="" type="checkbox"/> pdo_firebird Firebird PDO	
<input checked="" type="checkbox"/> pdo_ibm PDO IBM	Enable.
<input checked="" type="checkbox"/> pdo_informix Informix PDO	Enable.

Check the steps below to complete the manual installation of Scriptcase in a typical way.

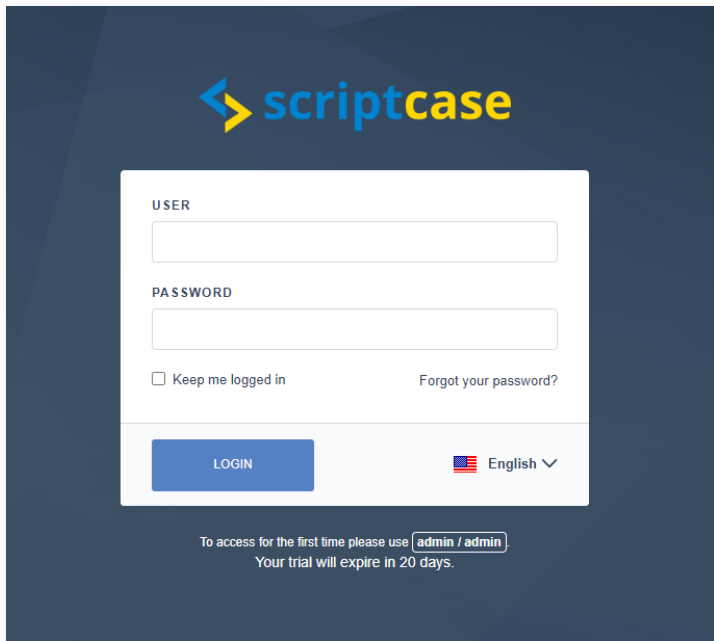
NOTE: By selecting this installation, Scriptcase will set your environment by default. Defining the installation database and the login user in Scriptcase.

Typical Installation:
Scriptcase will install everything automatically.

Customized Installation:
You could choose on which database you wish to install ScriptCase and the default username and password to access it.

[Next](#)

After clicking proceed, you will be redirected to the Scriptcase Login page.



User
In this field, it is necessary to inform the user to login to Scriptcase. By default, the user is **admin**.

Password
In this field, you must enter the password to login to Scriptcase. By default, the password is **admin**.

Keep me connected
This option keeps the user logged in until they log out through the Scriptcase interface.

Forgot your password?
By clicking this option, it is possible to recover the password of the informed user. An email will be sent to the user in question so they can access the recovery steps.

Important: You can only perform password recovery if you have configured your SMTP information. To access our documentation showing how to perform this configuration click [here](#).

Custom Installation

Check the steps below to complete the manual installation of Scriptcase in a custom manner.

NOTE: By selecting this installation, you will be responsible for setting up your environment. Defining the installation database and the login user for Scriptcase.

Typical Installation:
Scriptcase will install everything automatically.

Customized Installation:
You could choose on which database you wish to install ScriptCase and the default username and password to access it.

[Next](#)

1- Verify that the directories required by Scriptcase have the correct permissions. If any directory listed does not have the active check icon , correct permission for this directory.

Installation :: System Folders

STEPS

- Initiating
- PHP Extensions
- System Folders**
- Database
- ScriptCase Tables
- User
- Finishing

ScriptCase performs many operations with files. It is necessary that some directories have the write permission set. Make sure the ScriptCase's directories have the necessary authorization and correct the problem.

- ✓ C:/Apache24/htdocs/scriptcase/app/
- ✓ C:/Apache24/htdocs/scriptcase/conf/
- ✓ C:/Apache24/htdocs/scriptcase/backup/
- ✓ C:/Apache24/htdocs/scriptcase/dev/
- ✓ C:/Apache24/htdocs/scriptcase/dev/conf/grp/
- ✓ C:/Apache24/htdocs/scriptcase/dev/conf/scriptcase/
- ✓ C:/Apache24/htdocs/scriptcase/dev/conf/usr/
- ✓ C:/Apache24/htdocs/scriptcase/file/doc/
- ✓ C:/Apache24/htdocs/scriptcase/file/img/
- ✓ C:/Apache24/htdocs/scriptcase/log/
- ✓ C:/Apache24/htdocs/scriptcase/tmp/

Next >

Click proceed to proceed with the installation process .

2- Choose the database where the Scriptcase database will be installed. By default, Scriptcase recommends that it be installed with the **SQLite** database.

Installation :: Database

STEPS

- Initiating
- PHP Extensions
- System Folders
- Database**
- ScriptCase Tables
- User
- Finishing

ScriptCase stores its applications data in a database. You must choose the connection parameters for ScriptCase to be able to connect to this database.

DBMS: ✓

We recommend to use SQLite as your main ScriptCase database. Using SQLite, you ensure a greater reliability in ScriptCase features, including **Automatic Backup** of your projects and applications. If you want to install ScriptCase in another database, be aware that the Automatic Backup ScriptCase will not consider your projects and most of the data related to it.

Change Database

Next >

- You can change the default database by clicking the **Change Database** button. By clicking this option you will need to set the following options:

Installation :: Database

STEPS

- Initiating
- PHP Extensions
- System Folders
- Database**
- ScriptCase Tables
- User
- Finishing

ScriptCase stores its applications data in a database. You must choose the connection parameters for ScriptCase to be able to connect to this database.

DBMS: ✓

Server:

User:

Password:

Database:

We recommend to use SQLite as your main ScriptCase database. Using SQLite, you ensure a greater reliability in ScriptCase features, including **Automatic Backup** of your projects and applications. If you want to install ScriptCase in another database, be aware that the Automatic Backup ScriptCase will not consider your projects and most of the data related to it.

Next >

DBMS

In this option, you will select the Driver to connect to your database.

Server

In this option, you will inform the server to connect to the desired database.

User

You need to inform a user who has access to the database you want Scriptcase to be installed on.

Password

You must enter the authentication password corresponding to the entered user.

Database

You must enter the name of the database where Scriptcase will be installed.

Example:

`scriptcase`

Clique em **prosseguir** para avançar no processo de instalação.

3 - Verifique se a criação das tabelas da base de dados do Scriptcase ocorreu com sucesso. As tabelas serão marcadas com o ícone de sucesso  na lateral.

Installation :: ScriptCase Tables

STEPS

- Initiating
- PHP Extensions
- System Folders
- Database
- ScriptCase Tables
- User
- Finishing

ScriptCase uses tables on a database to store the applications development data.

- ✓ sc_lbprj Projects table
- ✓ sc_lbusu User Table.
- ✓ sc_lbgroup Groups table
- ✓ sc_lbuser_group Relationship table for users and groups
- ✓ sc_lbati User Status Table.
- ✓ sc_lbapl Application Table.
- ✓ sc_lbcmp Fields Table.
- ✓ sc_lblog Table of log schemes
- ✓ sc_lblog_apl Applications Backup Table.
- ✓ sc_lblog_cmp Backup Table of fields.
- ✓ sc_lbrep Data Dictionary Table.
- ✓ sc_lbrep_tables Data Dictionary (Tables) Table;
- ✓ sc_lbrep_fields Data Dictionary (Fields) Table;
- ✓ sc_lbconex Connections Table.
- ✓ sc_lbseas Session table
- ✓ sc_lbversao Project Version Table.
- ✓ sc_lbvt Events Table.
- ✓ sc_lbvt_evt Events Backup Table.
- ✓ sc_lbtrans Transactions Table.
- ✓ sc_lbtdo Task list table
- ✓ sc_lbmsg Table of messages between users

All ScriptCase tables are already created.

[Next >](#)

Click proceed to proceed with the installation process.

_4 - _ Set the default user who will have access to Scriptcase.

Installation :: User

STEPS

- Initiating
- PHP Extensions
- System Folders
- Database
- ScriptCase Tables
- User
- Finishing

The access to ScriptCase is made through a user/password system. To access it for the first time, a user will be created with admin privileges. This will allow you to log on the system and create new users.

Login:

Password:

Confirmation:

[Next >](#)

Login

In this option, you will enter the default Scriptcase User Login.

Password

In this option, you must enter the password that will correspond to the user informed.

Confirmation

You must repeat the previously entered password for successful user creation.

Click proceed to proceed with the installation process.

5 - Complete Scriptcase custom installation on your web server.

Installation :: Finishing

STEPS

- Initiating
- PHP Extensions
- System Folders
- Database
- ScriptCase Tables
- User
- Finishing

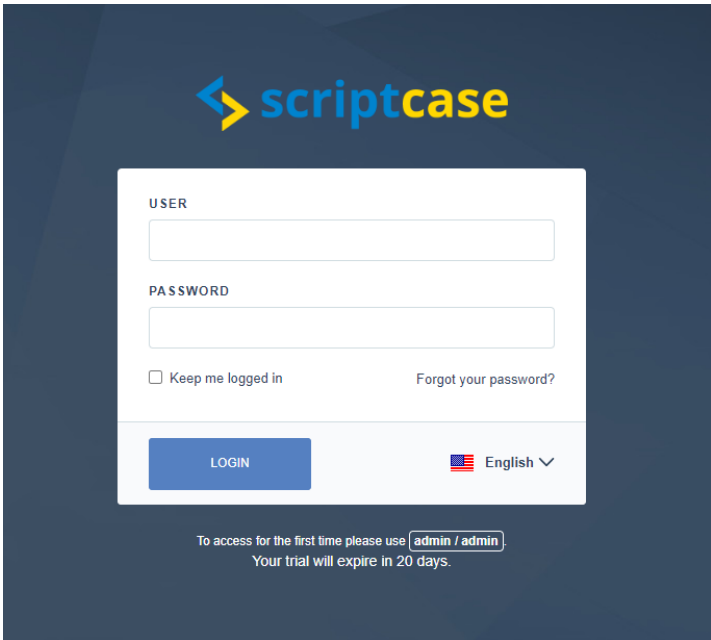
ScriptCase is now configured and ready to start.

Installation Complete.

Congratulations! ScriptCase was successfully installed on your web server. Click on the "Next" button to proceed to login page.

[Next >](#)

After clicking proceed, you will be redirected to the Scriptcase Login page.



User

In this field, it is necessary to inform the user to login to Scriptcase. By default, the user is **admin**.

Password

In this field, you must enter the password to login to Scriptcase. By default, the password is **admin**.

Keep me connected

This option keeps the user logged in until they log out through the Scriptcase interface.

Forgot your password?

By clicking this option, it is possible to recover the password of the informed user. An email will be sent to the user in question so they can access the recovery steps.

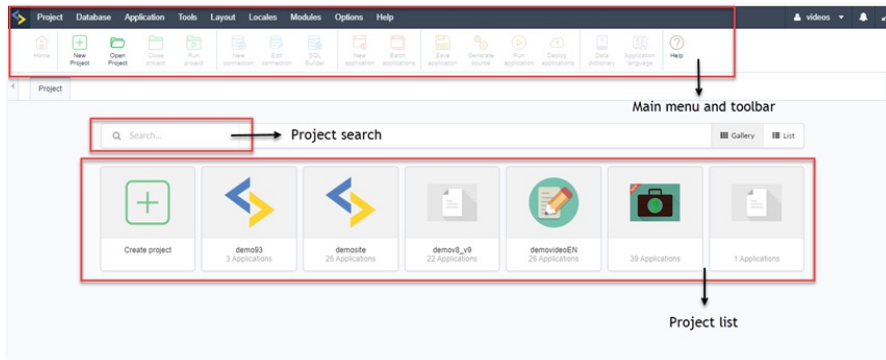
Important: You can only perform password recovery if you have configured your SMTP information. To access our documentation showing how to perform this configuration click [here](#).

Need to register your Scriptcase?
You can access our [knowledge base](#) to understand all the steps for registering a Scriptcase license.

In this article, we present to you the main options from Scriptcase's interface. There are two important screens: project dashboard and applications management interface.

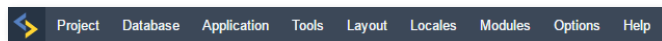
Project Dashboard

The project dash is the first interface you will see once you log into Scriptcase. It's where you start a new project or select an existing project for editing.



Main Menu

The main menu contains both Scriptcase's and project's general settings, and it is available at the top of the page. Most of the options are related to the project settings and are not available before the project creation.



Toolbar

The toolbar is displayed right below the main menu, where you can find the most frequent used functions used. The toolbar is customizable, and you can create new functions as well as edit the current ones. Learn more about the toolbar by clicking [here](#).



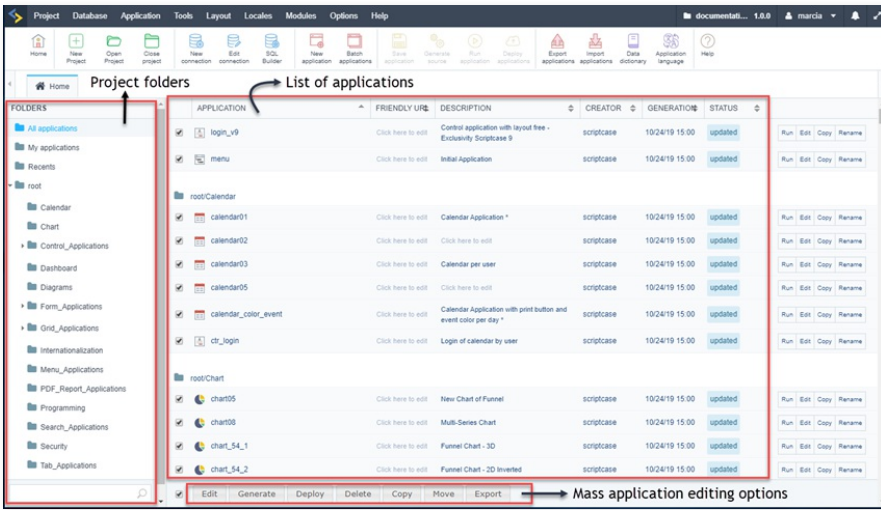
Project List

On this session is displayed a list with all the Scriptcase projects. You can display the project in a gallery or in list (to sort by date, name or creator). You have also a general search option.

+ Create project					
	Project	Description	Administrator	Created	Applications
	demo93	Novo Projeto	videos	2019-05-16	3
	demo93e	Novo Projeto	videos	2019-05-17	26
	demo93_y9	Novo Projeto	videos	2019-08-16	22
	demo93eEN	New Project	videos	2019-05-24	26
	fbdemo		videos	2017-08-02	39

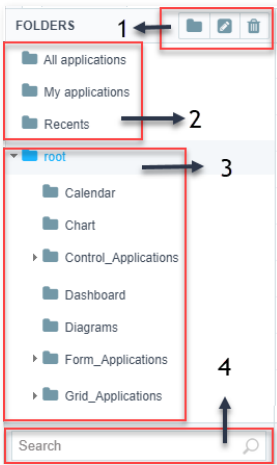
Application management interface

Once you create the project Scriptcase will display the applications management interface, where you can organize all project applications using folders. You are able also to search created applications, sort by date, creator, status, and take some actions such as run, generate, deploy, rename and copy. You can edit one or various applications at a time.



Project Explorer

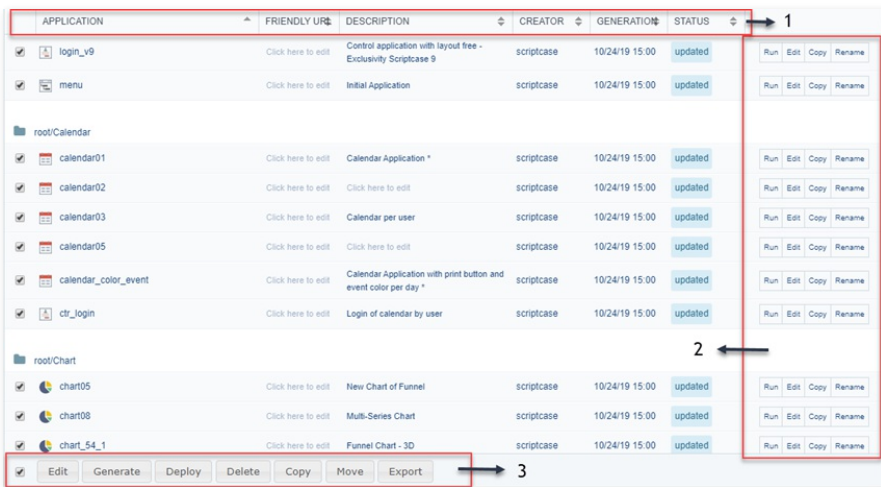
In this session, you can find all the folders created to organize the applications. There's also the folder that contains all the applications of the project in one place.



- - Folders functions;
- - Filters;
- - Folders' structure;
- - Application search.

List Of Applications

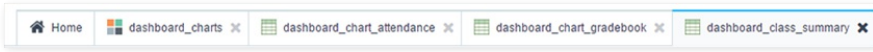
You can view a list of the applications that are in the selected folder in the Project Explorer presented on the left side.



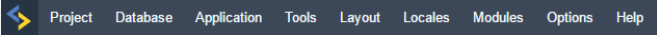
- - Columns;
- - Express options;
- - Footer Options.

Taskbar

This bar holds the tabs of all applications which are open for editing.



The main menu contains both Scriptcase's and project's general settings, and it is available at the top of the page. Most of the options are related to the project settings and are not available before the project creation. See below each item and a resume of its options:



Project

- **ew Project:** To create a new project or import one of the system templates.
- **pen Project:** Lists the projects that the user is permitted to access.
- **lose Project:** Closes the current project.
- **roperties:** Allows you to edit some of the project properties such as themes and languages.
- **efault Values:** Allows editing the current project default values.
- **ersion History:** Displays a list of project versions.
- **crement Version:** This option helps you to make the project version control.
- **xport Project:** Allows to export one project at a time, taking all the project's essential content for using in another Scriptcase of the same version.
- **mport Project:** Allows to import a project exported by Scriptcase of the same version.
- **enerate Source Code:** Generates the source code for all applications in the current project.
- **eploy Applications:** Opens the Publishing Wizard for publishing the applications of the current project.
- **delete Project:** This option is only available once you are into a project. This option deletes a project.
- **reports:**
 - **pplication List:** Displays the project applications with details of its creation.
 - **evelopers Summary:** Displays a summary of the applications and codes created by a developer.
- **how Diagram:** Displays a complete diagram of the project showing the relationship between the applications.
- **roject Search:** It allows to perform a search for texts in all the project's applications.

Database

- **ew Connection:** Allows you to create a connection in the current project.
- **dit Connection:** Allows you to edit existing connections in the current project.
- **mport ACCESS:** Allows the conversion of an ACCESS data source into a structured database (MySQL, PostgreSQL, SQLite, and SQL Server).
- **mport EXCEL:** Allows the conversion of an EXCEL data source into a structured database (MySQL, PostgreSQL, SQLite, and SQL Server).
- **mport CSV:** Allows the conversion of a CSV data source into a structured database (MySQL, PostgreSQL, SQLite, and SQL Server).
- **QL Builder:** Tool for creating and executing SQL commands.
- **atabase Builder:** Tool for rapidly creating and editing tables in the database.
- **R Diagram:** Tool that allows the visualization of the relationships between tables of the connected database.

Application

- **ew Application:** Create a new application.
- **atch Applications:** Allows simultaneous creation of Grids and Forms.
- **estore Applications:** Allows the restoration of applications, where is displayed a list of the last edited applications.
- **ave Application:** Save changes made to the current application.
- **enerate Source:** Generate the Source Code for all the applications in the current project.
- **un Application:** Saves and executes the current application.
- **xport Applications:** Allows the export of one or more project applications.
- **mport Applications:** Imports a backup file that contains an application from another or the same project.
- **ource Code:** Allows you to visualize the generated source code of the current application.
- **ata in Session:** It displays all Scriptcase session variables, as well as those of the executed applications.

Tools

- **ata Dictionary:** Allows the user to model and standardize the tables fields (names, translation, field types, and general settings).
- **xpress Edit:** Project applications mass editing.
- **elpCase:** Enables the creation of project documentation.
- **ternal Libraries:** It allows you to import third-party libraries from the web (javascript, CSS, PHP, for example) or to create your libraries to use within your current project or entire Scriptcase projects.
- **nternal Libraries:** Allows you to create PHP scripts, similar to external libraries, but exclusively for PHP.
- **o-Do List:** Enables you to create a task list.
- **essages:** Send messages among developers of the same Scriptcase installation.

- **converters:**
 - **ersion 6:** Converts your projects from ScriptCase version 6 to this current version.
 - **ersion 7:** Converts your projects from ScriptCase version 7 to this current version.
 - **ersion 8:** Converts your projects from ScriptCase version 8 to this current version.

Layout

- **SS Applications (Themes):** This option allows you to create or edit themes for the applications.
- **SS Buttons:** This option allows you to create or edit button themes.
- **TML Templates:** This option allows you to create or edit several HTML elements used within the applications.
- **SS Menus:** This option allows you to edit the CSS of the menus.
- **enu Icons:** Allows you to edit or create the icons of the menu application.
- **mage Manager:** This option manages the existing images in the project, also allowing you to add, delete and copy images.
- **hart Themes:** This option allows you to edit or create the themes used within the Chart applications.
- **TML Editor Templates:** Allows you to configure the functionalities from the Form and Control HTML editor field.

Local

- **pplication Language:** Allows to define custom messages in different languages, according to the ones set during the project creation and the data dictionary.
- **egional Settings:** Allows you to define some parameters of monetary units, date and numbers according to the Country or Region.

Modules

- **ecurity:** This option allows you to implement access rules to your system, creating a complete system with access control in your project.
- **og:**
 - **reate/Edit Log Module:** Allows you to create or edit a log module.
 - **pplications Related to the log:** Allows you to reference which applications will use the log.
 - **reate Query with Log Report:** Allows you to create a Grid application that will display a log report.

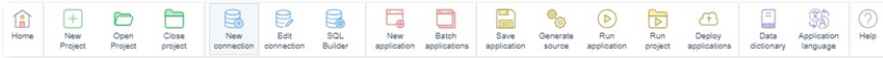
Options

- **ettings:** Access all the ScriptCase settings.
- **y ScriptCase:** Allows you to customize the settings for the current user.
- **hange Password:** Allows the current user to change their password.
- **y Toolbar:** Allows you to change the position of the icons on the toolbar.

Help

- **ebHelp:** Allows you to access the manual.
- **echnical Support:** Access to ScriptCase online support.
- **iagnosis:** Displays the settings of the Environment/Server where ScriptCase is installed.
- **heck Version:** Verifies if the version is updated.
- **pdate Version:** Allows you to update ScriptCase to the latest version available.
- **icense Registration:** Access the page to register your ScriptCase.
- **bout:** Information about ScriptCase.

The toolbar is displayed right below the main menu, where you can find the most frequent used functions used. The toolbar is customizable, and you can create new features as well as edit the current ones. Learn how to customize the toolbar by clicking [here](#).

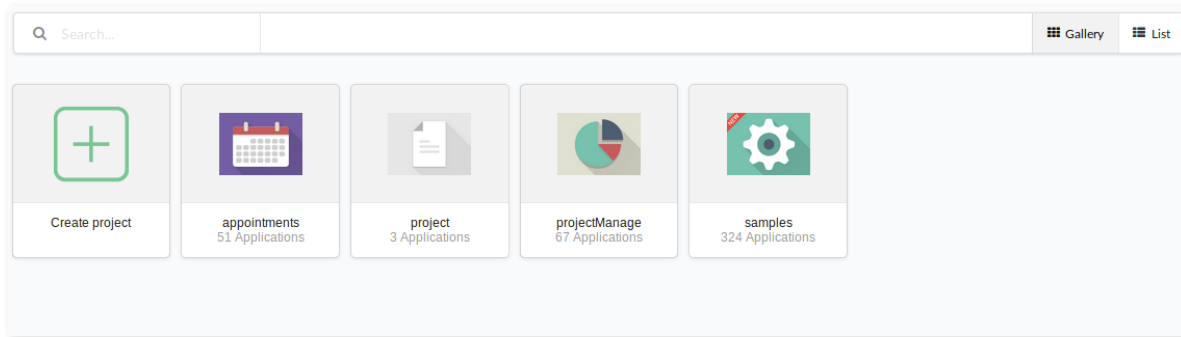


- **Home:** Returns to the Application management interface.
- **New Project:** Starts a new project creation process.
- **Open Project:** Returns to project dash interface where you can see all the projects that your current user has access.
- **Close Project:** Closes the current project.
- **New Connection:** Displays a wizard to add new connections to the current project.
- **Edit Connection:** Edits the project database connections.
- **SQL Builder:** SQL editor to create or test queries inside the connected database. You can also run SQL commands using this feature.
- **New Application:** It opens the new application wizard for you to start a new application.
- **Batch Applications:** Creates various applications (Forms and Grids) simultaneously.
- **Save Application:** Saves the current application changes.
- **Generate Source Code:** Generate the Source Code for the existing Applications.
- **Run Application:** Saves the present application, generates the source code, and runs the applications.
- **Run Project:** Runs the complete project. To do this, you must enter the property in the project properties.
- **Deploy Applications:** Opens the wizard for the deployment procedure.
- **Data Dictionary:** Allows the user to define standards for the tables (Names, Field Types, etc.).
- **Application Language:** Feature to assist within translation of the system tables and internal messagers.
- **Help:** Scriptcase's documentation.

In this area is displayed all the projects that the logged user is permitted to access.

There are two ways of visualizing projects. Gallery (default) or list.

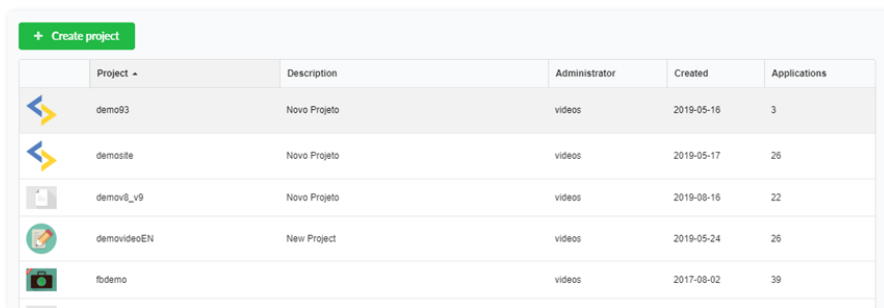
Project Gallery








The Project Gallery interface includes a search bar at the top left, a search icon, and a search input field. To the right of the search bar are two view toggles: 'Gallery' (selected) and 'List'. Below the search bar are five project cards, each with an icon and text:

- Create project**: A green plus sign icon.
- appointments**: 51 Applications. A purple calendar icon.
- project**: 3 Applications. A grey document icon.
- projectManage**: 67 Applications. A green and blue pie chart icon.
- samples**: 324 Applications. A green gear icon with a red 'NEW' tag.

Project List

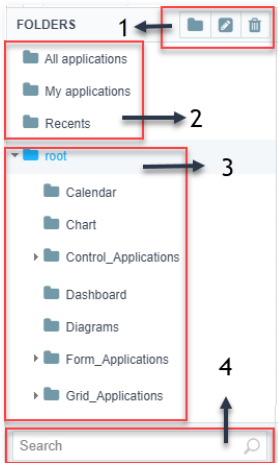


The Project List interface features a green '+ Create project' button at the top left. Below it is a table with the following data:

Project	Description	Administrator	Created	Applications
 demo93	Novo Projeto	videos	2019-05-16	3
 demosite	Novo Projeto	videos	2019-05-17	26
 demov0_v9	Novo Projeto	videos	2019-08-16	22
 demovideoEN	New Project	videos	2019-05-24	26
 fbdemo		videos	2017-08-02	39

To start a new project click on [Project > New Project](#).

In this area, you can find all the folders created to organize the applications.



1 - Folders functions

Options for you to manage the folders (create, rename delete) to organize your application folders.

- **reate:** Creates a new folder.
- **ename:** Renames the selected folder.
- **elete:** Deletes the selected folder.

2 - Filters

Using the filter, you can quickly find an application per date or user. You can also display all the applications.

- **ll Applications:** Lists all applications from the folders and the existing subfolders.
- **y Applications:** Lists all the applications from the folders and subfolders that the current user has created.
- **ecent:** List of all the new generated applications.

3 - Folders' structure

Scriptcase lists all application folders and sub-folders. It's recommendable for you to create folders to organize your project for future updates.

4 - Application search

Searches for the applications names or description.

- **earch:** It searches by name or descriptions of the project. The search occurs within the selected folder, and the search results are highlighted in yellow.

All project applications are available in this area. You can also take some specific actions using this interface, such as add a description, a friendly URL, rename, edit, copy, or run the application, one by one or various simultaneously.

The columns can be edited in *My Scriptcase* menu.

APPLICATION	FRIENDLY URL	DESCRIPTION	CREATOR	GENERATION#	STATUS	
<input checked="" type="checkbox"/> login_v9	Click here to edit	Control application with layout free - Exclusivity Scriptcase 9	scriptcase	10/24/19 15:00	updated	Run Edit Copy Rename
<input checked="" type="checkbox"/> menu	Click here to edit	Initial Application	scriptcase	10/24/19 15:00	updated	Run Edit Copy Rename
root/Calendar						
<input checked="" type="checkbox"/> calendar01	Click here to edit	Calendar Application *	scriptcase	10/24/19 15:00	updated	Run Edit Copy Rename
<input checked="" type="checkbox"/> calendar02	Click here to edit	Click here to edit	scriptcase	10/24/19 15:00	updated	Run Edit Copy Rename
<input checked="" type="checkbox"/> calendar03	Click here to edit	Calendar per user	scriptcase	10/24/19 15:00	updated	Run Edit Copy Rename
<input checked="" type="checkbox"/> calendar05	Click here to edit	Click here to edit	scriptcase	10/24/19 15:00	updated	Run Edit Copy Rename
<input checked="" type="checkbox"/> calendar_color_event	Click here to edit	Calendar Application with print button and event color per day *	scriptcase	10/24/19 15:00	updated	Run Edit Copy Rename
<input checked="" type="checkbox"/> ctr_login	Click here to edit	Login of calendar by user	scriptcase	10/24/19 15:00	updated	Run Edit Copy Rename
root/Chart						
<input checked="" type="checkbox"/> chart05	Click here to edit	New Chart of Funnel	scriptcase	10/24/19 15:00	updated	Run Edit Copy Rename
<input checked="" type="checkbox"/> chart08	Click here to edit	Multi-Series Chart	scriptcase	10/24/19 15:00	updated	Run Edit Copy Rename
<input checked="" type="checkbox"/> chart_54_1	Click here to edit	Funnel Chart - 3D	scriptcase	10/24/19 15:00	updated	Run Edit Copy Rename

1 - Columns

Here you have an editable grid with some suitable options from your applications, such as follows:

- **pplication**: Application's Name.
- **riendly URL**: Friendly URL for the application.
- **escription**: Application Description.
- **reator**: Application Creator.
- **eneration**: Last Source Code Generation date.
- **tatus**: Status of the application: **Updated** means that all changes are within the application, and **Outdated** means that changes are not saved within the source code.

2 - Express options

You can use these options to express editing the applications.

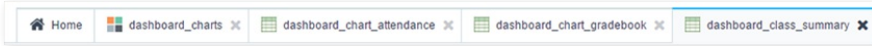
- **un**: Runs an application;
- **dit**: Open the Application to edit;
- **opy**: Creates a copy of the selected application;
- **ename**: Renames the selected application.

3 - Footer Options

The footer buttons are available when one or more application are selected.

- **dit**: Opens all the selected applications settings for editing.
- **enerate**: Generates the source code of all selected applications.
- **eploy**: Opens the Publishing Wizard for all selected applications.
- **delete**: Delete all selected applications.
- **opy**: Copy the selected applications to another project.
- **ove**: Moves the selected applications to another folder in the same project.
- **xport**: Allows you to export the selected applications.

This bar contains the tabs of all the applications that were opened for editing.

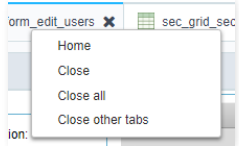


This bar is only displayed when at least one application has been opened.

The applications open in tabs in the taskbar where you can navigate among applications in a fast and easy way.

The "x" presented at the side of the Application name is used to close the application.







When you right-click on one of the tabs on the taskbar, you will see some tab closing options.



Keyboard hotkeys are keys or key combinations that provides an alternate way of doing something that you would normally do with a mouse.

The keyboard hotkeys are available in all browsers compatible with Scriptcase.

Browsers Compatible With Scriptcase

 Chrome	All versions
 Firefox	Version 4 or above
 Edge	All versions
 Internet Explorer	Version 11 or above
 Opera	Version 9 or above
 Safari	Version 5.2 or above

Scriptcase Hotkeys

Efficiency is crucial to software development and the Keyboard Hotkeys are important to achieve that efficiently. You can see all the hotkeys that are now available in Scriptcase below:

Hotkeys	Actions
F1	Opens the documentation of the tool
CTRL+H or ⌘+H or F4	Displays the project homepage
CTRL+S or ⌘+S or F7	Saves the changes made in the application
CTRL+B or ⌘+B or F8	Generates the source code of the current application
CTRL+E or ⌘+E or F9	Runs the current application
F2	Opens the application creation screen
CTRL+I or ⌘+I or F6	Access to the SQL Builder
Alt+1~9	Accesses the selected tabs 1~9
Alt+Q	Closes the current tab
Alt+PgDown	Navigates back to the previous tab
Alt+PgUp	Navigates to the next tab

Events Editor Hotkeys - PHP

Using Sublime shortcut patterns, Scriptcase provides a series of shortcuts to streamline its development.

Hotkeys	Actions
Tab	Indents the line or selection
Shift+Tab	Removes line indentation or selection
Ctrl+Shift+K	Deletes the current line
Ctrl+Left	Moves the cursor to the next word
Ctrl+Right	Moves the cursor to the previous word
Ctrl+L	Selects the entire current line
Ctrl+Shift+L	Adds the cursor to all selected lines
Ctrl+	Cancels multiple cursors and moves the cursor to the first instance
Ctrl+Enter	Inserts a line after the current one
Ctrl+Shift+Enter	Inserts a line before the current one
Ctrl+Shift+D	Selects the next occurrence of selected text with the cursor
Ctrl+Shift+Space	Selects the current scope (within parentheses, braces, brackets ...)
Ctrl+Shift+M	Selects the broader context (keys, brackets, parentheses, page)
Ctrl+M	Moves the cursor to the closing or opening of the current context. (braces, brackets, parentheses, tags)
Ctrl+/	Comments or Uncomments line
Ctrl+J	Groups the line or selection
Ctrl+D	Duplicates line or selection
Ctrl+F9	Sorts the lines alphabetically, ignoring upper cases
Backspace	Intelligent Backspace (Removes indentation with line)
Ctrl+K Ctrl+K	Deletes text to the right
Ctrl+K Ctrl+U	Capital letters in the selection or cursor
Ctrl+K Ctrl+L	Lowercase letters in the selection or cursor
Ctrl+K Ctrl+C	Scrolls the line to the center of view
Ctrl+K Ctrl+Backspace	Deletes text to the left
Ctrl+Shift+Up	Adds the cursor to the previous line
Ctrl+Shift+Down	Adds the cursor to the next line
Ctrl+F3	Finds the next occurrence of the selection or cursor
Ctrl+Shift+F3	Finds the next occurrence of the selection or cursor
Ctrl+Shift+[Shows indented tag
Ctrl+Shift+]	Shows Inline tag
Ctrl+Shift+F	Activates search and replaces
Ctrl+F	Activates search
F3	Finds the next occurrence of the search
Shift+F3	Finds the previous occurrence of the search
Ctrl+Alt+I	Auto indentation of the selected code

Events Editor Hotkeys - HTML/CSS

Using Emmet's shortcut patterns, Scriptcase provides a series of shortcuts to streamline its development. With this plug-in it is possible to write HTML and CSS codes through CSS selectors.

[Click Here](#) to view the list of selectors.

For example, just write `ul> li.item $ {This is the item $} * 3` then press the "TAB" key and the following code will be generated.

```
<li class="item1">This is item 1</li>
<li class="item2">This is item 2</li>
<li class="item3">This is item 3</li>
</ul>
```

To use the EMMET hotkeys in Scriptcase events, you must close and open the event PHP. Example: `?> ul> li.item $ {This is the item $} * 3 <? PHP`

Hotkeys	Actions
Tab	Enables EMMET abbreviations Ex. <code>ul>li.item\${item \$}*3</code>
Ctrl+Shift+A or ⌘+Shift+A	Envelopes the selected text in an EMMET abbreviation. Select the text you want to envelop, then use the keys, and enter the tag.
Ctrl+Alt+Right or ⌘+Alt+Right	Edit / selects the next node. Junction between tags, CSS blocks.
Ctrl+Alt+Left or ⌘+Alt+Left	Edit / selects the previous node. Junction between tags, CSS blocks.
Ctrl+Shift+Y or ⌘+Shift+Y	Validates and solves math expressions. Example: <code>2*8+4 .</code>
Ctrl+Up or ⌘+Up	Increments numeric value <code>+1</code>
Ctrl+Down or ⌘+Down	Decrements numerical value <code>-1</code>
Ctrl+Shift+. or ⌘+Shift+.	Edit / selects the next item. Attribute, tag name
Ctrl+Shift+, or ⌘+Shift+,	Edit / selects the previous item. Attribute, tag name
Enter	Inserts indented line break.

Field Variables

In Scriptcase the fields are internally treated as [local variables](#) PHP, however, at Scriptcase interface it's necessary to inform these, using **keys**, so it's possible to retrieve or assign values to the application fields.

These fields can be used at Scriptcase events, but will not work from the **onScriptnit** event because only the events after that will have values in the fields.

In the examples below it's possible to visualize the operation of the fields.

Retrieving field value:

To assign the field value to a local variable to be used within an event, we can perform the steps as below:

```
$var_local = {application_field};
```

To assign the field value to a local variable to be used within an event, we can perform the steps as below:

```
{variable_global} = {application_field};
```

Assigning Value to Field:

To assign a value to the field, we can do assignment in the same way as done with the variables, and if possible, according to the field type:

Field type Text: {field_text} = "This is a sample text";

Field type Number: {field_number} = 100;

Global Variables (Session Variables)

These are variables that are stored in the application session, these variables can be used to pass parameters through the applications in the project. These variables can be used at any application event.

Unlike the local and field variables, Global variables can be used in the SQL of the applications that allows the manual to be changed, and in the **WHERE** Clause of the Form application, thus enabling a dynamic use of the applications.

SQL Grid sample

```
SELECT
  orderid,
  customerid,
  employeaid,
  orderdate,
  requireddate,
  shippeddate,
  shipvia,
  freight,
  priceorder,
  shipcountry,
  shipregion,
  shipstate,
  shipcity,
  shipname,
  shipaddress,
  shippostalcode
FROM
  orders
WHERE orderid = [order]
```

Sample in the WHERE Clause of the Form

Where clause

To define a global variable, you only have to enter it in between brackets, for example:

```
[global_variable].
```

Assigning value to global variable:

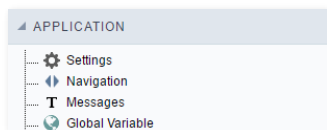
[text_variable] = "This global has a text stored in it";

[number_variable] = "This global has a number stored in it";

Passing values between applications

To pass values between applications, it is necessary to define the same variable in the applications that will use them, however, in the source application the variable must be defined as **Output** and in the destination application must be defined as **Input**.

To change the variable type, go to the menu **Application> Global Variables**.



After clicking on this menu will show the variables in the application and its settings.

VARIABLE SETTINGS	
ATTRIBUTE	VALUE
global	<div style="border: 1px solid #ccc; padding: 5px;"> Scope <ul style="list-style-type: none"> <input type="checkbox"/> SESSION <input checked="" type="checkbox"/> POST <input checked="" type="checkbox"/> GET </div> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 5px;"> Settings <ul style="list-style-type: none"> <input type="checkbox"/> Optional </div> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 5px;"> Type <ul style="list-style-type: none"> <input type="radio"/> Out <input checked="" type="radio"/> In </div>

In this configuration screen we have the following configuration options:

Attribute:

In Attributes we can see the variables in our application.

Value:

In Value we can see some options and configurations of our variable.

Type: This option defines whether the variable will be input or output.

Description:

Reports where the variable was declared by the developer.

Local Variables

Local variables are basically PHP's own variables. These variables must be set using a **\$** dollar sign at the beginning and the variable names are case-sensitive.

These variables will only work in events or methods, using a variable in an event it will make it only available in that event.

Variable names follow the same rules as other labels in PHP. A valid variable name begins with a letter or underscore, followed by any number of letters, numbers, or underscores.

Below we can see some examples of local variables:

\$text = "This variable stores a Text";

\$number = "This variable stores a number";

For more information, visit the [php variables](#).

Below we have a list of EMMET CSS selectors in our event editor.

Remember that as they are CSS Selectors, to work as expected it is necessary to use inside the tag `<style> @m </style>`

CSS selectors	To open
@i or @import	@import url();
@m or @media	@media screen { }
@f	@font-face {font-family: ; src:url(); }
@f+	@font-face {font-family: 'FontName'; src: url('FileName.eot'); src: url('FileName.eot?#iefix') format('embedded-opentype'), url('FileName.woff') format('woff'), url('FileName.ttf') format('truetype'), url('FileName.svg#FontName') format('svg'); font-style: normal; font-weight: normal; }
anim	animation: ;
anim-	animation: name duration timing-function delay iteration-count direction fill-mode;
animdel	animation-delay: time;
animdir or animdir:n	animation-direction: normal;
animdir:r	animation-direction: reverse;
animdir:a	animation-direction: alternate;
animdir:ar	animation-direction: alternate-reverse;
animdur	animation-duration: 0s;
animfm	animation-fill-mode: both;
animfm:f	animation-fill-mode: forwards;
animfm:b	animation-fill-mode: backwards;
animfm:bt or animfm:bh	animation-fill-mode: both;
animic	animation-iteration-count: 1;
animici	animation-iteration-count: infinite;
animn	animation-name: none;
animps	animation-play-state: running;
animps:p	animation-play-state: paused;
animps:r	animation-play-state: running;
animtf	animation-timing-function: linear;
animtfe	animation-timing-function:ease;
animtfei	animation-timing-function:ease-in;
animtfeo	animation-timing-function:ease-out;
animtfeio	animation-timing-function:ease-in-out;
animtfl	animation-timing-function:linear;
animtfc	animation-timing-function: cubic-bezier(0.1, 0.7, 0.1, 0.1);
ap	appearance: none;
!	!important"
pos or pos:r	position: relative;
pos:s	position:static;

pos:a CSS selectors	<code>position:absolute;</code> To open
pos:f	<code>position:fixed;</code>
t	<code>top;</code>
t:a	<code>top:auto;</code>
r	<code>right;</code>
r:a	<code>right:auto;</code>
b	<code>bottom;</code>
b:a	<code>bottom:auto;</code>
l	<code>left;</code>
l:a	<code>left:auto;</code>
z	<code>z-index;</code>
z:a	<code>z-index:auto;</code>
fl or fl:l	<code>float:left;</code>
fl:n	<code>float:none;</code>
fl:r	<code>float:right;</code>
cl or cl:b	<code>clear:both;</code>
cl:n	<code>clear:none;</code>
cl:l	<code>clear:left;</code>
cl:r	<code>clear:right;</code>
colm	<code>columns;</code>
colmc	<code>column-count;</code>
colmf	<code>column-fill;</code>
colmg	<code>column-gap;</code>
colmr	<code>column-rule;</code>
colmrc	<code>column-rule-color;</code>
colmrs	<code>column-rule-style;</code>
colmrw	<code>column-rule-width;</code>
colms	<code>column-span;</code>
colmw	<code>column-width;</code>
d or d:b	<code>display:block;</code>
d:n	<code>display:none;</code>
d:f	<code>display:flex;</code>
d:if	<code>display:inline-flex;</code>
d:i	<code>display:inline;</code>
d:ib	<code>display:inline-block;</code>
d:ib+	<code>display:inline-block;\n*display:inline;\n*zoom:1;</code>
d:li	<code>display:list-item;</code>
d:ri	<code>display:run-in;</code>
d:cp	<code>display:compact;</code>
d:tb	<code>display:table;</code>
d:itb	<code>display:inline-table;</code>

css selectors	To open
d:tbl	<code>display:table-caption;</code>
d:tbl	<code>display:table-column;</code>
d:tblg	<code>display:table-column-group;</code>
d:tblhg	<code>display:table-header-group;</code>
d:tblfg	<code>display:table-footer-group;</code>
d:tbr	<code>display:table-row;</code>
d:tbrg	<code>display:table-row-group;</code>
d:tblc	<code>display:table-cell;</code>
d:rb	<code>display:ruby;</code>
d:rbb	<code>display:ruby-base;</code>
d:rbbg	<code>display:ruby-base-group;</code>
d:rbrt	<code>display:ruby-text;</code>
d:rbrtg	<code>display:ruby-text-group;</code>
v or vh	<code>visibility: hidden;</code>
v:v	<code>visibility:visible;</code>
v:c	<code>visibility:collapse;</code>
ov or ov:h	<code>overflow: hidden;</code>
ov:v	<code>overflow:visible;</code>
ov:s	<code>overflow:scroll;</code>
ov:a	<code>overflow:auto;</code>
ovx or ovx:h	<code>overflow-x: hidden;</code>
ovx:v	<code>overflow-x:visible;</code>
ovx:s	<code>overflow-x:scroll;</code>
ovx:a	<code>overflow-x:auto;</code>
ovy	<code>overflow-y: hidden;</code>
ovy:v	<code>overflow-y:visible;</code>
ovy:h	<code>overflow-y:hidden;</code>
ovys	<code>overflow-y:scroll;</code>
ovy:a	<code>overflow-y:auto;</code>
ovs or ovs:s	<code>overflow-style: scrollbar;</code>
ovs:a	<code>overflow-style:auto;</code>
ovs:p	<code>overflow-style:panner;</code>
ovs:m	<code>overflow-style:move;</code>
ovs:mq	<code>overflow-style:marquee;</code>
zoo	<code>zoom:1;</code>
zm	<code>zoom:1;</code>
cp	<code>clip: ;</code>
cp:a	<code>clip:auto;</code>
cp:r	<code>clip: rect(top right bottom left);</code>
bxz	<code>box-sizing: border-box;</code>
bxz:cb	<code>box-sizing:content-box;</code>

CSS selectors	To open
border	border: border-box;
bxsh	box-shadow: inset hoff voff blur color;
bxsh:r	box-shadow: inset hoff voff blur spread rgb(0, 0, 0);
bxsh:ra	box-shadow: inset h v blur spread rgba(0, 0, 0, .5);
bxsh:n	box-shadow:none;
m	margin: ;
m:a	margin:auto;
mt	margin-top: ;
mt:a	margin-top:auto;
mr	margin-right: ;
mr:a	margin-right:auto;
mb	margin-bottom: ;
mb:a	margin-bottom:auto;
ml	margin-left: ;
ml:a	margin-left:auto;
p	padding: ;
pt	padding-top: ;
pr	padding-right: ;
pb	padding-bottom: ;
pl	padding-left: ;
w	width: ;
w:a	width:auto;
h	height: ;
h:a	height:auto;
maw	max-width: ;
maw:n	max-width:none;
mah	max-height: ;
mah:n	max-height:none;
miw	min-width: ;
mih	min-height: ;
mar	max-resolution: res;
mir	min-resolution: res;
ori	orientation: ;
ori:l	orientation:landscape;
ori:p	orientation:portrait;
ol	outline: ;
ol:n	outline:none;
olo	outline-offset: ;
olw	outline-width: ;
olw:tn	outline-width:thin;
olw:m	outline-width:medium;

CSS selectors	To open
ol:w:tc	<code>outline-width:thick;</code>
ols	<code>outline-style: ;</code>
ols:n	<code>outline-style:none;</code>
ols:dt	<code>outline-style:dotted;</code>
ols:ds	<code>outline-style:dashed;</code>
ols:s	<code>outline-style:solid;</code>
ols:db	<code>outline-style:double;</code>
ols:g	<code>outline-style:groove;</code>
ols:r	<code>outline-style:ridge;</code>
ols:i	<code>outline-style:inset;</code>
ols:o	<code>outline-style:outset;</code>
olc	<code>outline-color: #000;</code>
olci	<code>outline-color:invert;</code>
bfv	<code>backface-visibility: ;</code>
bfv:h	<code>backface-visibility:hidden;</code>
bfv:v	<code>backface-visibility:visible;</code>
bd	<code>border: ;</code>
bd+	<code>border: 1px solid #000;</code>
bd:n	<code>border:none;</code>
bdbk or bdbk:c	<code>border-break: close;</code>
bdcl	<code>border-collapse: ;</code>
bdcl:c	<code>border-collapse:collapse;</code>
bdcl:s	<code>border-collapse:separate;</code>
bdc	<code>border-color: #000;</code>
bdc:t	<code>border-color:transparent;</code>
bdi	<code>border-image: url();</code>
bdi:n	<code>border-image:none;</code>
bdti	<code>border-top-image:url();</code>
bdti:n	<code>border-top-image:none;</code>
bdri	<code>border-right-image:url();</code>
bdri:n	<code>border-right-image:none;</code>
bdbi	<code>border-bottom-image:url();</code>
bdbi:n	<code>border-bottom-image:none;</code>
bdli	<code>border-left-image:url();</code>
bdli:n	<code>border-left-image:none;</code>
bdci	<code>border-corner-image:url();</code>
bdci:n	<code>border-corner-image:none;</code>
bdci:c	<code>border-corner-image:continue;</code>
bdtli	<code>border-top-left-image:url();</code>
bdtli:n	<code>border-top-left-image:none;</code>

css selectors	To open
bdli	<code>border-top-left-image:continue;</code>
bdtri	<code>border-top-right-image:url();</code>
bdtri:n	<code>border-top-right-image:none;</code>
bdtri:c	<code>border-top-right-image:continue;</code>
bdbri	<code>border-bottom-right-image:url();</code>
bdbri:n	<code>border-bottom-right-image:none;</code>
bdbri:c	<code>border-bottom-right-image:continue;</code>
bdbli	<code>border-bottom-left-image:url();</code>
bdbli:n	<code>border-bottom-left-image:none;</code>
bdbli:c	<code>border-bottom-left-image:continue;</code>
bdf or bdfr	<code>border-fit: repeat;</code>
bdf:c	<code>border-fit:clip;</code>
bdf:sc	<code>border-fit:scale;</code>
bdf:st	<code>border-fit:stretch;</code>
bdf:ow	<code>border-fit:overwrite;</code>
bdf:of	<code>border-fit:overflow;</code>
bdf:sp	<code>border-fit:space;</code>
bdlen	<code>border-length: ;</code>
bdlen:a	<code>border-length:auto;</code>
bdsp	<code>border-spacing:;</code>
bds	<code>border-style: ;</code>
bds:n	<code>border-style:none;</code>
bds:h	<code>border-style:hidden;</code>
bds:dt	<code>border-style:dotted;</code>
bds:ds	<code>border-style:dashed;</code>
bds:s	<code>border-style:solid;</code>
bds:db	<code>border-style:double;</code>
bds:dt ds	<code>border-style:dot-dash;</code>
bds:dt dt ds	<code>border-style:dot-dot-dash;</code>
bds:w	<code>border-style:wave;</code>
bds:g	<code>border-style:groove;</code>
bds:r	<code>border-style:ridge;</code>
bds:i	<code>border-style:inset;</code>
bds:o	<code>border-style:outset;</code>
bdw	<code>border-width: ;</code>
bdtw	<code>border-top-width: ;</code>
bdrw	<code>border-right-width: ;</code>
bdbw	<code>border-bottom-width: ;</code>
bdlw	<code>border-left-width: ;</code>
bdt or bt	<code>border-top: ;</code>
bdt+	<code>border-top: 1px solid #000;</code>

CSS selectors	To open
border-top	<code>border-top:none;</code>
border-top-style	<code>border-top-style: ;</code>
border-top-style:none	<code>border-top-style:none;</code>
border-top-color	<code>border-top-color: #000;</code>
border-top-color:transparent	<code>border-top-color:transparent;</code>
border-right or border-right	<code>border-right: ;</code>
border-right: 1px solid #000	<code>border-right: 1px solid #000;</code>
border-right:none	<code>border-right:none;</code>
border-right-style	<code>border-right-style: ;</code>
border-right-style:none	<code>border-right-style:none;</code>
border-right-color	<code>border-right-color: #000;</code>
border-right-color:transparent	<code>border-right-color:transparent;</code>
border-bottom or border-bottom	<code>border-bottom: ;</code>
border-bottom: 1px solid #000	<code>border-bottom: 1px solid #000;</code>
border-bottom:none	<code>border-bottom:none;</code>
border-bottom-style	<code>border-bottom-style: ;</code>
border-bottom-style:none	<code>border-bottom-style:none;</code>
border-bottom-color	<code>border-bottom-color: #000;</code>
border-bottom-color:transparent	<code>border-bottom-color:transparent;</code>
border-left or border-left	<code>border-left: ;</code>
border-left: 1px solid #000	<code>border-left: 1px solid #000;</code>
border-left:none	<code>border-left:none;</code>
border-left-style	<code>border-left-style: ;</code>
border-left-style:none	<code>border-left-style:none;</code>
border-left-color	<code>border-left-color: #000;</code>
border-left-color:transparent	<code>border-left-color:transparent;</code>
border-radius	<code>border-radius: ;</code>
border-top-right-radius	<code>border-top-right-radius: ;</code>
border-top-left-radius	<code>border-top-left-radius: ;</code>
border-bottom-right-radius	<code>border-bottom-right-radius: ;</code>
border-bottom-left-radius	<code>border-bottom-left-radius: ;</code>
background	<code>background: #000;</code>
background: #fff url() 0 0 no-repeat	<code>background: #fff url() 0 0 no-repeat;</code>
background:none	<code>background:none;</code>
background-image: url('x.png')	<code>filter:progid:DXImageTransform.Microsoft.AlphaImageLoader(src='x.png',sizingMethod='crop');</code>
background-color	<code>background-color: #fff;</code>
background-color:transparent	<code>background-color:transparent;</code>
background-image:url()	<code>background-image:url();</code>
background-image:none	<code>background-image:none;</code>
background-repeat	<code>background-repeat: ;</code>
background-repeat:no-repeat	<code>background-repeat:no-repeat;</code>

CSS selectors bgr:x	To open background-repeat:repeat-x;
bgr:y	background-repeat:repeat-y;
bgr:sp	background-repeat:space;
bgr:rd	background-repeat:round;
bga	background-attachment ;
bga:f	background-attachment:fixed;
bga:s	background-attachment:scroll;
bgp	background-position: 0 0;
bgpx	background-position-x: ;
bgpy	background-position-y: ;
bgbk	background-break: ;
bgbk:bb	background-break:bounding-box;
bgbk:eb	background-break:each-box;
bgbk:c	background-break:continuous;
bgcp or bgcp:pb	background-clip: padding-box;
bgcp:bb	background-clip:border-box;
bgcp:cb	background-clip:content-box;
bgcp:nc	background-clip:no-clip;
bgo	background-origin: ;
bgo:pb	background-origin:padding-box;
bgo:bb	background-origin:border-box;
bgo:cb	background-origin:content-box;
bgsz	background-size: ;
bgsz:a	background-size:auto;
bgsz:ct	background-size:contain;
bgsz:cv	background-size:cover;
c	color: #000;
cr	color: rgb(0, 0, 0);
cra	color: rgba(0, 0, 0, .5);
cm	/* */
cnt	content:' ';
cnt:n	content:normal;
cnt:oq	content:open-quote;
cnt:noq	content:no-open-quote;
cnt:cq	content:close-quote;
cnt:ncq	content:no-close-quote;
cnt:a	content:attr();
cnt:c	content:counter();
cnt:cs	content:counters();
tbl	table-layout: ;

Class selectors	To open
tbla	<code>table-layout:auto;</code>
tblf	<code>table-layout:fixed;</code>
cps	<code>caption-side: ;</code>
cps:t	<code>caption-side:top;</code>
cps:b	<code>caption-side:bottom;</code>
ec	<code>empty-cells: ;</code>
ecs	<code>empty-cells:show;</code>
ech	<code>empty-cells:hide;</code>
lis	<code>list-style: ;</code>
lis:n	<code>list-style:none;</code>
lisp	<code>list-style-position: ;</code>
lisp:i	<code>list-style-position:inside;</code>
lisp:o	<code>list-style-position:outside;</code>
list	<code>list-style-type: ;</code>
list:n	<code>list-style-type:none;</code>
listd	<code>list-style-type:disc;</code>
listc	<code>list-style-type:circle;</code>
lists	<code>list-style-type:square;</code>
listdc	<code>list-style-type:decimal;</code>
listdclz	<code>list-style-type:decimal-leading-zero;</code>
listlr	<code>list-style-type:lower-roman;</code>
listur	<code>list-style-type:upper-roman;</code>
lisi	<code>list-style-image: ;</code>
lisi:n	<code>list-style-image:none;</code>
q	<code>quotes: ;</code>
q:n	<code>quotes:none;</code>
q:ru	<code>quotes: '00AB' '00BB' '201E' '201C';</code>
q:en	<code>quotes: '201C' '201D' '2018' '2019';</code>
ct	<code>content: ;</code>
ct:n	<code>content:normal;</code>
ct:oq	<code>content:open-quote;</code>
ct:noq	<code>content:no-open-quote;</code>
ct:cq	<code>content:close-quote;</code>
ct:ncq	<code>content:no-close-quote;</code>
cta	<code>content:attr();</code>
ct:c	<code>content:counter();</code>
ct:cs	<code>content:counters();</code>
coi	<code>counter-increment: ;</code>
cor	<code>counter-reset: ;</code>
va or vat	<code>vertical-align: top;</code>
va:sup	<code>vertical-align:super;</code>

CSS selectors	To open
va:t	vertical-align:top;
va:m	vertical-align:middle;
va:bl	vertical-align:baseline;
va:b	vertical-align:bottom;
va:tb	vertical-align:text-bottom;
va:sub	vertical-align:sub;
ta or ta:l	text-align: left;
ta:c	text-align:center;
ta:r	text-align:right;
ta:j	text-align:justify;
ta-lst	text-align-last: ;
ta:la	text-align-last:auto;
ta:ll	text-align-last:left;
ta:lc	text-align-last:center;
ta:lr	text-align-last:right;
td or td:n	text-decoration: none;
td:u	text-decoration:underline;
td:o	text-decoration:overline;
td:l	text-decoration:line-through;
te	text-emphasis: ;
te:n	text-emphasis:none;
te:ac	text-emphasis:accent;
te:dt	text-emphasis:dot;
te:c	text-emphasis:circle;
te:ds	text-emphasis:disc;
te:b	text-emphasis:before;
te:a	text-emphasis:after;
th	text-height: ;
th:a	text-height:auto;
th:f	text-height:font-size;
th:t	text-height:text-size;
th:m	text-height:max-size;
ti	text-indent: ;
ti:-	text-indent:-9999px;
tj	text-justify: ;
tj:a	text-justify:auto;
tj:iw	text-justify:inter-word;
tj:ii	text-justify:inter-ideograph;
tj:ic	text-justify:inter-cluster;
tj:d	text-justify:distribute;
tj:k	text-justify:kashida;

CSS selectors	To open
tj:t	text-justify:tibetan;
to: or tove:	text-overflow: ellipsis;
to:c	text-overflow:clip;
to	text-outline: ;
to+	text-outline: 0 0 #000;
to:n	text-outline:none;
tr	text-replace: ;
tr:n	text-replace:none;
tt	text-transform: uppercase;
tt:n	text-transform:none;
tt:c	text-transform:capitalize;
tt:u	text-transform:uppercase;
tt:l	text-transform:lowercase;
tw	text-wrap: ;
tw:n	text-wrap:normal;
tw:no	text-wrap:none;
tw:u	text-wrap:unrestricted;
tw:s	text-wrap:suppress;
tsh	text-shadow: hoff voff blur #000;
tshr	text-shadow: h v blur rgb(0, 0, 0);
tsh:ra	text-shadow: h v blur rgba(0, 0, 0, .5);
tsh+	text-shadow: 0 0 0 #000;
tsh:n	text-shadow:none;
trf	transform: ;
trf:skx	transform: skewX(angle);
trf:sky	transform: skewY(angle);
trf:sc	transform: scale(x, y);
trf:scx	transform: scaleX(x);
trf:scy	transform: scaleY(y);
trf:scz	transform: scaleZ(z);
trf:sc3	transform: scale3d(x, y, z);
trf:r	transform: rotate(angle);
trf:rx	transform: rotateX(angle);
trf:ry	transform: rotateY(angle);
trf:rz	transform: rotateZ(angle);
trf:t	transform: translate(x, y);
trf:tx	transform: translateX(x);
trf:ty	transform: translateY(y);
trf:tz	transform: translateZ(z);
trf:t3	transform: translate3d(tx, ty, tz);
trfo	transform-origin: ;

trf	transform-style: preserve-3d;
trfs	transform-style: preserve-3d;
trs	transition: prop time;
trsde	transition-delay:time;
trsdu	transition-duration:time;
trsp	transition-property:prop;
trstf	transition-timing-function:tfunc;
lh	line-height: ;
whs	white-space: ;
whs:n	white-space: normal;
whs:p	white-space: pre;
whs:nw	white-space: nowrap;
whs:pw	white-space: pre-wrap;
whs:pl	white-space: pre-line;
whsc	white-space-collapse: ;
whsc:n	white-space-collapse:normal;
whsc:k	white-space-collapse:keep-all;
whsc:l	white-space-collapse:loose;
whsc:bs	white-space-collapse:break-strict;
whsc:ba	white-space-collapse:break-all;
wob	word-break: ;
wob:n	word-break:normal;
wob:k	word-break:keep-all;
wob:ba	word-break:break-all;
wos	word-spacing: ;
wow	word-wrap: ;
wow:nm	word-wrap:normal;
wow:n	word-wrap:none;
wow:u	word-wrap:unrestricted;
wow:s	word-wrap:suppress;
wow:b	word-wrap:break-word;
wm or wm:lrt	writing-mode:lr-tb;
wm:lrb	writing-mode:lr-bt;
wm:rlt	writing-mode:rl-tb;
wm:rlb	writing-mode:rl-bt;
wm:tbr	writing-mode:tb-rl;
wm:tbl	writing-mode:tb-lr;
wm:btl	writing-mode:bt-lr;
wm:btr	writing-mode:bt-rl;
lts	letter-spacing: ;
lts-n	letter-spacing:normal;

f	<code>font: ;</code>
CSS selectors	To open
f+	<code>font: 1em Arial,sans-serif;</code>
fw	<code>font-weight: ;</code>
fw:n	<code>font-weight:normal;</code>
fw:b	<code>font-weight:bold;</code>
fw:br	<code>font-weight:bolder;</code>
fw:lr	<code>font-weight:lighter;</code>
fs or fs:i	<code>font-style:italic;</code>
fs:n	<code>font-style:normal;</code>
fs:o	<code>font-style:oblique;</code>
fv	<code>font-variant: ;</code>
fv:n	<code>font-variant:normal;</code>
fv:sc	<code>font-variant:small-caps;</code>
fz	<code>font-size: ;</code>
fza	<code>font-size-adjust: ;</code>
fza:n	<code>font-size-adjust:none;</code>
ff	<code>font-family: ;</code>
ff:s	<code>font-family:serif;</code>
ff:ss	<code>font-family:sans-serif;</code>
ff:c	<code>font-family:cursive;</code>
ff:f	<code>font-family:fantasy;</code>
ff:m	<code>font-family:monospace;</code>
ff:a	<code>font-family: Arial, "Helvetica Neue", Helvetica, sans-serif;</code>
ff:t	<code>font-family: "Times New Roman", Times, Baskerville, Georgia, serif;</code>
ff:v	<code>font-family: Verdana, Geneva, sans-serif;</code>
fef	<code>font-effect: ;</code>
fef:n	<code>font-effect:none;</code>
fef:eg	<code>font-effect:engrave;</code>
fef:eb	<code>font-effect:emboss;</code>
fef:o	<code>font-effect:outline;</code>
fem	<code>font-emphasize: ;</code>
femp	<code>font-emphasize-position: ;</code>
femp:b	<code>font-emphasize-position:before;</code>
femp:a	<code>font-emphasize-position:after;</code>
fems	<code>font-emphasize-style: ;</code>
fems:n	<code>font-emphasize-style:none;</code>
fems:ac	<code>font-emphasize-style:accent;</code>
fems:dt	<code>font-emphasize-style:dot;</code>
fems:c	<code>font-emphasize-style:circle;</code>
fems:ds	<code>font-emphasize-style:disc;</code>
fsm	<code>font-smooth: ;</code>

fsm:a CSS selectors	font-smooth:auto; To open
fsm:n	font-smooth:never;
fsm:aw	font-smooth:always;
fst	font-stretch: ;
fst:n	font-stretch:normal;
fst:uc	font-stretch:ultra-condensed;
fst:ec	font-stretch:extra-condensed;
fst:c	font-stretch:condensed;
fst:sc	font-stretch:semi-condensed;
fst:se	font-stretch:semi-expanded;
fst:e	font-stretch:expanded;
fst:ee	font-stretch:extra-expanded;
fst:ue	font-stretch:ultra-expanded;
op	opacity: ;
op+	opacity: ; filter: alpha(opacity=);
op:ie	filter:progid:DXImageTransform.Microsoft.Alpha(Opacity=100);
op:ms	-ms-filter:'progid:DXImageTransform.Microsoft.Alpha(Opacity=100)';
rsz	resize: ;
rsz:n	resize:none;
rsz:b	resize:both;
rsz:h	resize:horizontal;
rsz:v	resize:vertical;
cur	cursor:pointer;
cur:a	cursor:auto;
cur:d	cursor:default;
cur:c	cursor:crosshair;
cur:ha	cursor:hand;
cur:he	cursor:help;
cur:m	cursor:move;
cur:p	cursor:pointer;
cur:t	cursor:text;
fxd	flex-direction: ;
fxd:r	flex-direction:row;
fxd:rr	flex-direction:row-reverse;
fxd:c	flex-direction:column;
fxd:cr	flex-direction:column-reverse;
fxw	flex-wrap: ;
fxw:n	flex-wrap:nowrap;
fxw:w	flex-wrap:wrap;
fxw:wr	flex-wrap:wrap-reverse;
fix	flex-flow: ;

CSS selectors	To open
jc:fs	justify-content:flex-start;
jc:fe	justify-content:flex-end;
jc:c	justify-content:center;
jc:sb	justify-content:space-between;
jc:sa	justify-content:space-around;
ai	align-items: ;
ai:fs	align-items:flex-start;
ai:fe	align-items:flex-end;
ai:c	align-items:center;
ai:b	align-items:baseline;
ai:s	align-items:stretch;
ac	align-content: ;
ac:fs	align-content:flex-start;
ac:fe	align-content:flex-end;
ac:c	align-content:center;
ac:sb	align-content:space-between;
ac:sa	align-content:space-around;
ac:s	align-content:stretch;
ord	order: ;
fxg	flex-grow: ;
fxsh	flex-shrink: ;
fixb	flex-basis: ;
fx	flex: ;
as	align-self: ;
as:a	align-self:auto;
as:fs	align-self:flex-start;
as:fe	align-self:flex-end;
as:c	align-self:center;
as:b	align-self:baseline;
as:s	align-self:stretch;
pgbb	page-break-before: ;
pgbb:au	page-break-before:auto;
pgbb:al	page-break-before:always;
pgbb:l	page-break-before:left;
pgbb:r	page-break-before:right;
pgbi	page-break-inside: ;
pgbi:au	page-break-inside:auto;
pgbi:av	page-break-inside:avoid;
pgba	page-break-after: ;
pgba:au	page-break-after:auto;

CSS selectors	To open
page-break-after	<code>page-break-after:always;</code>
page-break-after:left	<code>page-break-after:left;</code>
page-break-after:right	<code>page-break-after:right;</code>
orphans	<code>orphans: ;</code>
user-select	<code>user-select:none;</code>
widows	<code>widows: ;</code>
wfsm or wfsm:a	<code>-webkit-font-smoothing:antialiased;</code>
wfsm:s or wfsm:sa	<code>-webkit-font-smoothing:subpixel-antialiased;</code>
wfsm:n	<code>-webkit-font-smoothing:none;</code>

HTML Selectors	To open
!!!	<code><!DOCTYPE html></code>
!!!4t	<code><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd"></code>
!!!4s	<code><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd"></code>
!!!xt	<code><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"></code>
!!!xs	<code><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"></code>
!!!xbs	<code><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd"></code>
c	<code><!-- --></code>
cc:ie6	<code><!--[if lte IE 6]> <![endif]--></code>
cc:ie	<code><!--[if IE]> <![endif]--></code>
cc:noie	<code><!--[if !IE]><!--><!--<![endif]--></code>
!	Create an HTML5 structure
a	<code></code>
a:link	<code></code>
a:mail	<code></code>
abbr	<code><abbr title=""></abbr></code>
acr or acronym	<code><acronym title=""></acronym></code>
base	<code><base href=""></code>
basefont	<code><basefont></code>
br	<code>
</code>
frame	<code><frame></code>
hr	<code><hr></code>
bdo	<code><bdo dir=""></bdo></code>
bdo:r	<code><bdo dir="rtl"></bdo></code>
bdo:l	<code><bdo dir="ltr"></bdo></code>
col	<code><col></code>
link	<code><link rel="stylesheet" href=""></code>
link:css	<code><link rel="stylesheet" href="style.css"></code>
link:print	<code><link rel="stylesheet" href="print.css" media="print"></code>
link:favicon	<code><link rel="shortcut icon" type="image/x-icon" href="favicon.ico"></code>

link:touch HTML Selectors	<link rel="apple-touch-icon" href="favicon.png"> To open
link:rss	<link rel="alternate" type="application/rss+xml" title="RSS" href="rss.xml">
link:atom	<link rel="alternate" type="application/atom+xml" title="Atom" href="atom.xml">
link:im or link:import	<link rel="import" href="component.html">
meta	<meta>
meta:utf	<meta http-equiv="Content-Type" content="text/html;charset=UTF-8">
meta:win	<meta http-equiv="Content-Type" content="text/html;charset=windows-1251">
meta:vp	<meta name="viewport" content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
meta:compat	<meta http-equiv="X-UA-Compatible" content="IE=7">
style	<style>
script	<script></script>
script:src	<script src=""></script>
img	
img:s or img:srcset	
img:z or img:sizes	
picture	<picture>
src or source	<source>
src:sc or source:src	<source src="" type="">
src:s or source:srcset	<source srcset="">
src:m or source:media	<source media="(min-width:)" srcset="">
src:t or source:type	<source srcset="" type="image/">
src:z or source:sizes	<source sizes="" srcset="">
src:mt or source:media:type	<source media="(min-width:)" srcset="" type="image/">
src:mz or source:media:sizes	<source media="(min-width)" sizes="" srcset="">
src:zt or source:sizes:type	<source sizes="" srcset="" type="image/">
iframe	<iframe src="" frameborder="0">
embed	<embed src="" type="">
object	<object data="" type="">
param	<param name="" value="">
map	<map name="">
area	<area shape="" coords="" href="" alt="">
area:d	<area shape="default" href="" alt="">
area:c	<area shape="circle" coords="" href="" alt="">
area:r	<area shape="rect" coords="" href="" alt="">
area:p	<area shape="poly" coords="" href="" alt="">
form	<form action="">
form:get	<form action="" method="get">
form:post	<form action="" method="post">
label	<label for="">
input	<input type="text">
inp	<input type="text" name="" id="">

input:h or input:hidden HTML Selectors	<code><input type="hidden" name=""></code> To open
input:t or input:text	<code><input type="text" name="" id=""></code>
input:s	<code><input type="search" name="" id=""></code>
input:e	<code><input type="email" name="" id=""></code>
input:u	<code><input type="url" name="" id=""></code>
input:p or input:password	<code><input type="password" name="" id=""></code>
input:d	<code><input type="datetime" name="" id=""></code>
input:da	<code><input type="date" name="" id=""></code>
input:da-l	<code><input type="datetime-local" name="" id=""></code>
input:m	<code><input type="month" name="" id=""></code>
input:w	<code><input type="week" name="" id=""></code>
input:ti	<code><input type="time" name="" id=""></code>
input:tel	<code><input type="tel" name="" id=""></code>
input:n	<code><input type="number" name="" id=""></code>
input:c	<code><input type="color" name="" id=""></code>
input:c or input:checkbox	<code><input type="checkbox" name="" id=""></code>
input:r or input:radio	<code><input type="radio" name="" id=""></code>
input:r	<code><input type="range" name="" id=""></code>
input:f or input:file	<code><input type="file" name="" id=""></code>
input:s or input:submit	<code><input type="submit" value=""></code>
input:i or input:image	<code><input type="image" src="" alt=""></code>
input:b or input:button	<code><input type="button" value=""></code>
isindex	<code><isindex></code>
input:r	<code><input type="reset" value=""></code>
select	<code><select name="" id=""></select></code>
select:d or select:disabled	<code><select name="" id="" disabled></select></code>
opt or option	<code><option value=""></option></code>
textarea	<code><textarea name="" id="" cols="30" rows="10"></textarea></code>
marquee	<code><marquee behavior="" direction=""></marquee></code>
menu:c or menu:context	<code><menu type="context"></menu></code>
menu:t or menu:toolbar	<code><menu type="toolbar"></menu></code>
video	<code><video src=""></video></code>
audio	<code><audio src=""></audio></code>
html:xml	<code><html xmlns="http://www.w3.org/1999/xhtml"></html></code>
keygen or kg	<code><keygen></code>
command or cmd	<code><command></code>
btn:s	<code><button type="submit"></button></code>
btn:r	<code><button type="reset"></button></code>
btn:d	<code><button disabled></button></code>
fst:d	<code><fieldset disabled></fieldset></code>
bq	<code><blockquote></blockquote></code>

HTML Selectors	To open </figure>
figcaption	<figcaption></figcaption>
pic	<picture></picture>
ifr	<iframe src="" frameborder="0"></iframe>
emb	<embed src="" type="">
obj	<object data="" type=""></object>
cap	<caption></caption>
colg	<colgroup></colgroup>
fst or fset	<fieldset></fieldset>
btn	<button></button>
optg	<optgroup></optgroup>
textarea	<textarea name="" id="" cols="30" rows="10"></textarea>
leg	<legend></legend>
sect	<section></section>
art	<article></article>
hdr	<header></header>
ft	<footer></footer>
adr	<address></address>
dlg	<dialog></dialog>
str	
prog	<progress></progress>
mn	<main></main>
tem	<template></template>
datag	<datagrid></datagrid>
datal	<datalist></datalist>
out	<output></output>
det	<details></details>
doc	Create an structure in HTML5 with the title "Documento"
doc4	Create an structure in HTML5 with the http-equiv="Content-Type" content="text/html;charset=UTF-8"
ri:d or ri:dpr	
ri:v or ri:viewport	
ri:a or ri:art	Create an structure with the following tags '<picture>', '<source media="(min-width:)" srcset="">' e ''
ri:t or ri:type	Create an structure with the following tags '<picture>', '<source srcset="" type="image/">' e ''
html:4t	Create an structure in HTML4 using 'loose.dtd'
html:4s	Create an structure in HTML4 using 'strict.dtd'
html:xt	Create an structure in HTML4 using 'xhtml1-transitional.dtd'
html:xs	Create an structure in HTML4 using 'xhtml1-strict.dtd'
html:xxs	Create an structure in HTML4 using 'xhtml11.dtd'
html:5	Create an structure in HTML5
ol+	Create an structure with the following tags '' e ''
ul+	Create an structure with the following tags '' e ''

HTML Selectors	To open
dl+	Create an structure with the following tags '<dl>', '<dt>' e '<dd>'
map+	Create an structure with the following tags '<map name="">' e '<area shape="" coords="" href="" alt="">'
table+	Create an structure with the following tags '<table>', '<tr>' e '<td>'
colgroup+ or colg+	Create an structure with the following tags '<colgroup>' e '<col>'
tr+	Create an structure with the following tags '<tr>' e '<td>'
select+	Create an structure with the following tags '<select name="" id="">' e '<option value=""></option>'
optgroup+ or optg+	Create an structure with the following tags '<optgroup>' e '<option value=""></option>'
pic+	Create an structure with the following tags '<picture>', '<source srcset="">' e ''

Scriptcase uses the concept of projects to manage the development of the applications. You will use the projects to organize your system development in a simple and practical way.

In a project, it's possible to define the applications development standards using the [default values](#), work with project [version control](#), [deploy](#) all the applications or part of them, create developers work flow [reports](#), display application [diagrams](#). Each project from Scriptcase can have 3 privilege levels for users or user groups. These privileges can be set inside the menu [options](#).

To create a Project on Scriptcase, it is necessary to have an interaction with a [Database](#). This way you can create applications based on database tables and views.

It's recommended to review your database tables and relationships to ensure that you have them correctly structured and organized. This will make your work on creating applications easier and faster when performing fields lookups and etc. You can visualize the current database structure through the menu > Database > Diagram ER.

[Click here](#) to see the steps to create a project.

RELATED LINKS

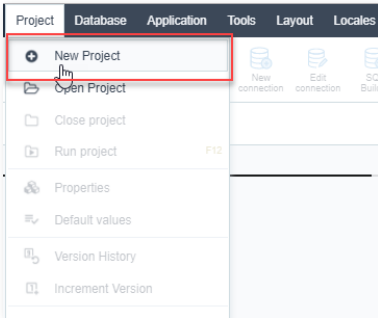
[Data Modeling and Project Creation](#)

[Scriptcase 9 Presentation](#)

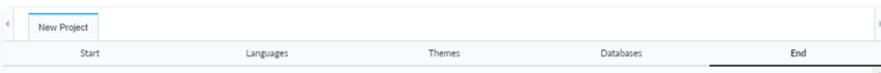
In this article you learn how to create a new project within Scriptcase.

A system made with Scriptcase has always to start by creating a project. A project for Scriptcase is a set of applications with some standard settings. During the project creation, you define some essential options as name and description, languages in which your project might be available, a layout template for the applications, and informs the database system connection data in which the applications fetch the information to form the system.

To start a new project you can access the menu "Project > New Project."



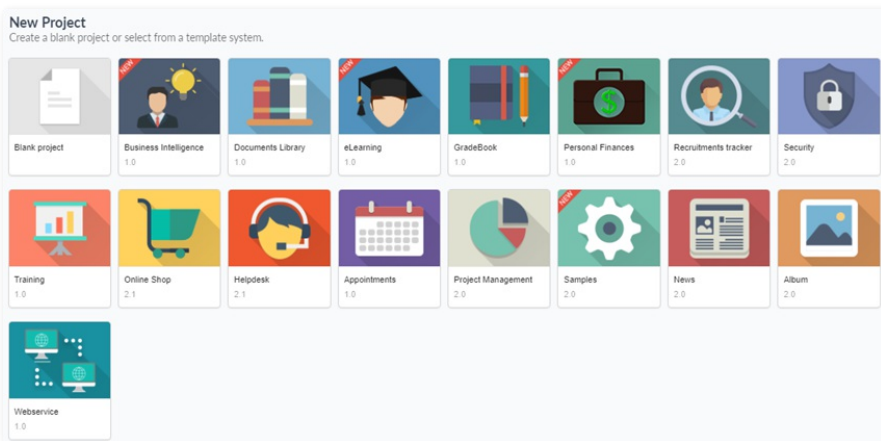
The project creation process has five steps, which are: **Start, Languages, Themes, Database and End.**



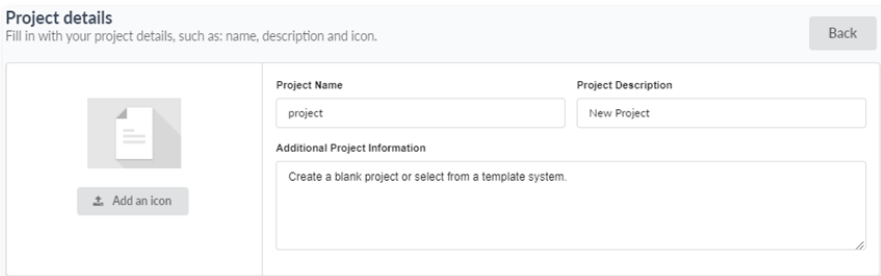
Start

In this step, we should choose between a **Blank project**, where we have to create our database connection and every application from scratch, or one select of our **Template projects**, that can serve as a template to the creation of your system.

[See more details on our Sample Projects and how to import them .](#)



After we select **Blank project**, it is mandatory to define a name to our project.



Project name

The only mandatory setting for this step is that the name must contain 1 to 32 alphanumeric characters and does not allow it to start with a number.

Project description

A not required that allows us to define a description for the project.

After the creation of the project, we can see the description in the dialog box when we place the mouse over the project when we use the gallery view or in the Description column when the project's view is in **list** format.

Additional Project Information

Not required fields that allow us to add additional information to the project. This information is within the project properties.

Image

Not required fields that allow us to define an image to represent our project.

You can change all non-mandatory fields in this step, and other project settings after project creation by accessing the option [Project Properties](#).

Languages

Here we choose the language for the project applications, and we must set at least one language for the project creation.







Language

In this step, you should select the languages that are available for system development.

Scriptcase can easily create multilingual systems. You need to add more than one language in this step of the project. All the languages you choose are available to the project, and you must choose one as standard. You can also have the same language more than once using different regional settings.

Scriptcase will automatically translate and apply the regional settings to your project applications (for buttons and internal messages for example). You can also use another tool from Scriptcase called [Data dictionary](#) to import and translate your database fields. You will be able to see, create or customize the messages inside the menu > Locales > [Application language](#).

Project languages
Choose the languages that you will use for your applications. It is possible to choose regional settings and define a charset for each selected language.

Language	Regional Settings	Charset	Default	Delete
 English	English (United States) ▼	Western (ISO-8859-1) ▼	<input checked="" type="radio"/>	
 Spanish	Spanish (Spain) ▼	Western (ISO-8859-15) ▼	<input type="radio"/>	
 Portuguese Portugal	Portuguese (Portugal) ▼	Western (ISO-8859-15) ▼	<input type="radio"/>	

Select a language ▼ Select the regional setting ▼ [Add](#)

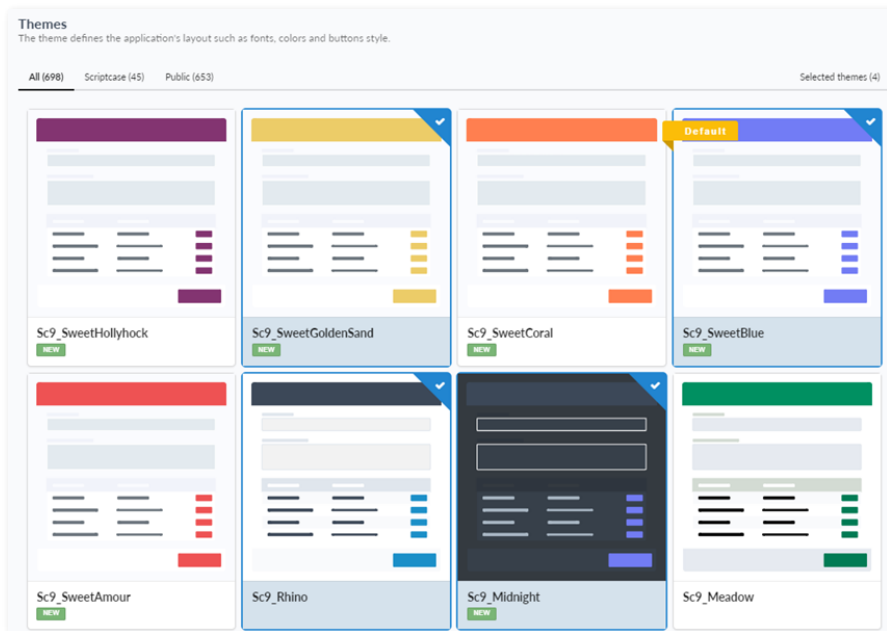
- **anguage** - List of languages that are in the projects.
- **egional Settings** - Contains the standard [regional settings](#) for the selected language. Regional Settings allows you to define some parameters of monetary units, dates, and numbers according to the country or region where your application.
- **harset** - Sets the character encoding format for the application. This encoding must be the same used in your database. By default, the charset comes according to the language, and you can change after creation in project properties.
- **efault** - Sets the project default language. Project applications are performed and displayed using this default language; it can also be changed individually within the application settings.
- **elete** - Removes the language of the project. In this way, the language removed will no longer be available. It may also be included or deleted after the project created within the Menu > Project > Properties.

After selecting the language, regional setting, and charset, click "Add" and then "Save" to move forward.

Themes

In the themes selection, you can choose one more than templates for your project. Within this template you will have a complete theme for all applications you create within the project. Ready to use screens with pre made CSS and HTML to all application types.

You can also customize or create new themes from the templates. Once you are inside the projects, access the menu [Layout > CSS Application \(Themes\)](#).



- **ll** - Lists all the available themes for use in your project.
- **criptcase** - Lists all Scriptcase standards themes.
- **ublic** - Lists all themes defined in the Public level.
- **ected themes** - Lists the themes selected for use in the project.

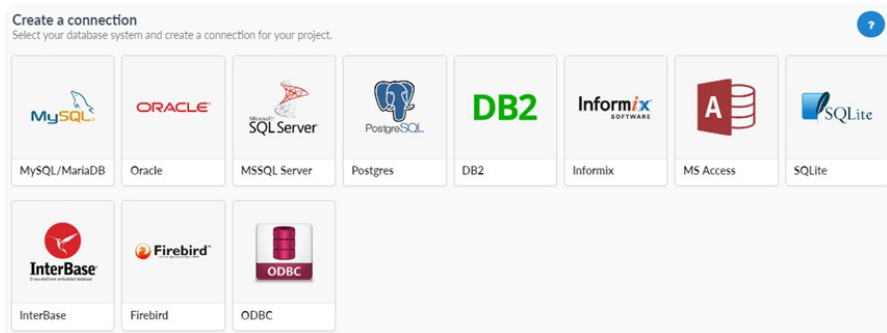
To add a theme to your project, you only need to select it, and then it is going to be listed inside the selected themes. You should select a theme to use as default, for this click on the “default” icon of the selected theme. The default theme is the one selected every time you create a new application, and you can also select a different one according to the options available within the selected themes. You can also access this option in the future to add new themes.

When you add more than one theme, you can dynamically switch themes before creating the applications, by adding a toolbar button within the applications or using the macro [sc_set_theme](#).

Databases

During the connection creation, Scriptcase lists all database systems according to your license. If you’re evaluating, Scriptcase displays all database systems available. You have to select your database and proceed to the next step to inform the database connection details.

You need to have a database system installed and the permissions to connect before you proceed.



The information required to connect is different for each database. In this example we will be using MySQL. To learn more about other connections please access [this link](#).

Using Scriptcase you are able to create as well as to manage database projects and tables. During this step you will need to create or connect to, at least, one database project. We are going to use a “Samples” one. After the project creation you are able to use the [Database manager](#) and [SQL Builder](#) features to manage the database project and queries.

Create a connection MySQL/MariaDB
Enter the connection details for your database MySQL/MariaDB

[← Choose another connection](#)

CONNECTION SECURITY FILTER ADVANCED ?

Connection Name: DBMS Driver:

Server/Host (Name or IP): Port:

Username: Password:

Database Name:

Database connection details.


- **Connection Name:** Define the name of the connection. You are not able to change this name in the future.
- **BMS Driver :** Defined as MySQL PDO. It can also be MySQLi.
- **Server/Host (Name or IP):** Enter the IP of the server where the database is. If the database is on the same machine of Scriptcase, you can use the IP **127.0.0.1** or **localhost**. If the Database is
- **Port_:** Define the port of the connection. By default, the port is 3306.
- **ser:** Inform the username that you use to connect to your database.
- **assword:** Inform the password that you use to connect to your database.
- **atabase Name:** Click on the "List Database" and select the desired database.
- **est Connection:** Displays the connection status if it succeeded or not.

All available connections are listed according to on your license type. [Go to our on-line store](#) and view the available licenses.

If you have connection problems, [access our free support](#) to get assistance by chat or remote access.

End

The project has been created successfully with the basic settings, and now you can create the applications. Visit the [Applications](#) menu to learn more about each application and to see the steps to create them.



Congratulations! Your project has been created successfully. You can now create Forms, Reports, Charts, Pivot Tables, among others applications to this project.

USEFUL LINKS

[Samples](#)
[Videos](#)
[Forum](#)
[Support](#)
[Documentation](#)
[Courses](#)

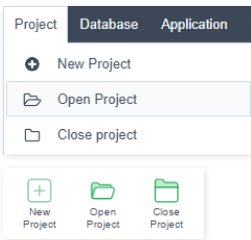
RELATED LINKS

[Data Modeling and Project Creation](#)

[Scriptcase 9 Presentation](#)

This option redirects to the Project List page, without closing your current project. To return to the application list, just click on the home icon. If you access another project, your current project is closed

Access **Project > Open Project** or click on the Open Project icon on the toolbar.

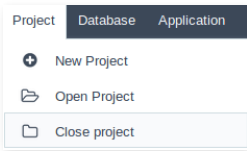


Next, you will visualize the project list. The current project appears on the upper right corner, as shown in image below:

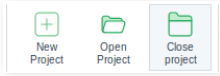
							PROJECT				
							elearning 1.0.0				
APPLICATION	FRIENDLY URL	DESCRIPTION	CREATOR	GENERATION	STATUS						
Classes											
Communication											
Instructor											
Security											
Student											
<input type="checkbox"/> blank	Home	Click here to edit	scriptcase	01/17/18 11:09	updated	Run	Edit	Copy	Rename		
<input type="checkbox"/> container	Click here to edit	Click here to edit	scriptcase	01/17/18 11:09	updated	Run	Edit	Copy	Rename		
<input type="checkbox"/> container_admin	Click here to edit	Click here to edit	scriptcase	01/17/18 11:09	updated	Run	Edit	Copy	Rename		
<input type="checkbox"/> container_instructor	Click here to edit	Click here to edit	scriptcase	01/17/18 11:09	updated	Run	Edit	Copy	Rename		
<input type="checkbox"/> form_conf_email	Click here to edit	Email Configuration	scriptcase	01/17/18 11:09	updated	Run	Edit	Copy	Rename		
<input type="checkbox"/> login	Click here to edit	HTML Login	scriptcase	01/17/18 11:09	updated	Run	Edit	Copy	Rename		
<input type="checkbox"/> menu	Click here to edit	Click here to edit	scriptcase	01/17/18 11:09	updated	Run	Edit	Copy	Rename		
<input type="checkbox"/> tabs_last_comunic	Click here to edit	Click here to edit	scriptcase	01/17/18 11:09	updated	Run	Edit	Copy	Rename		

Closes the current project redirecting to the the Project List.

Access **Project > Close Project**



Or simple click on the Close Project Icon on the toolbar.



This option is only available when accessing a project. It allows the editing of the project's information.

Edit Project

We can edit some of the project information, like the description and the use of the index.

- **project Name** - The name of the project informed on its creation. It is not possible to change the project name after its creation.
- **project Description** - Description informed on the project creation. Differently from the project name, it is possible to change it using this interface.
- **choose the initial application** - Allows you to define the initial project application for the **Run Project** button. So every time you run the project, this application opens automatically, it can be a login screen, for example.
- **use Index Page** - Allows the use of friendly URLs in the applications. Scriptcase enables this option by default.
- **project Information** - Detailed description of the project.

Locales

Define the languages for the project applications. The languages are select as you create your project, you can edit this option to add or remove languages anytime.

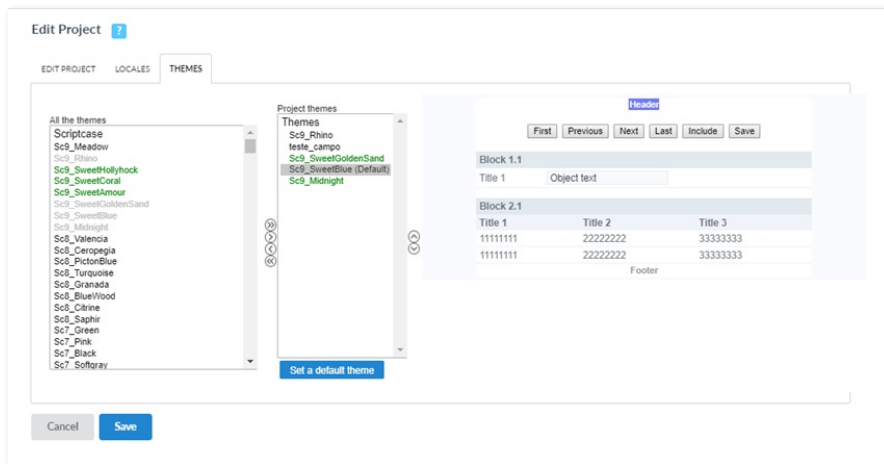
Language	Regional Settings	Charset	Default	Delete
English	English (United States)	Unicode (UTF-8)	<input checked="" type="radio"/>	
Portuguese Brazil	Portuguese (Brazil)	Unicode (UTF-8)	<input type="radio"/>	

Russian

- **language** - List of languages from the project.
- **regional Settings** - Standard regional settings for the selected language. Regional Settings allows you to define some parameters of monetary units, dates, and numbers according to the country or region where your application.
- **charset** - Encoding used by the selected language.
- **default** - Default a language for the project. All the applications created are executed on the default selected language.
- **trash Can** - Removes the language from the project. The removed language won't be available for changing dynamically in runtime.

Themes


It defines the available themes in the project and the default theme used by the applications. To set a default theme for your project, select the theme and click on **Set a default Theme**, the “default” notation appears beside the theme name. In this image, the default theme is **Sc9_Rhino**.



- **All Themes** - List of all the themes from ScriptCase.
- **Project Themes** - List of available themes within the project.



With this configuration, it is possible to standardize the creation of the applications of the current project, increasing productivity in the development of the applications.

Common Settings

Logo	scriptcase_NM_scriptcase5_logo_simple.png 
Records per page	30
Display Line Number	<input type="radio"/> Yes <input checked="" type="radio"/> No
Display Titles	<input checked="" type="radio"/> Yes <input type="radio"/> No
Show Summary	<input checked="" type="radio"/> Yes <input type="radio"/> No
Use the old quick search	<input checked="" type="radio"/> Yes <input type="radio"/> No
Use old Chart theme palettes	<input checked="" type="radio"/> Yes <input type="radio"/> No
Initial sort fields	
text	Ascendant ▼
date	Ascendant ▼
number	Ascendant ▼
currency	Ascendant ▼

- **Logo** - The Favicon that is going to use with the project. Customize the project using the same Favicon in all the applications.
- **Records per page** - Amount of records per page.
- **Display Line Number** - Defines the display of the line number existing in each application.
- **Display Titles** - Displays the title of the application in execution.
- **Show Summary** - Displays the number of records on the page and the total of the records in the application. This option executes only if the line option in the Grid toolbar is disabled.
- **Use the old quick search** - Defines the use of the old quick search, where we can specify which field do we want it to search.
- **Use old Chart theme palettes** - Defines the use of the old scheme of colors on the charts
- **Initial sort fields** - Defines the initial sorting of the fields.

Grid

Table Width	100
Table Width Unit	Percent ▼
Format Row Counter	{lang_othr_smry_info}
Table a Group By	10px
Separates the Group By	
Records per page	
TEMPLATES	
Header	Flat ▼ 
Footer	Default ▼ 
GRID FIELDS ALIGNMENT	
Horizontal Alignment	Default: Left ▼ Numeric: Right ▼ Date: Center ▼
Vertical Alignment	Top ▼ Top ▼ Top ▼
Title Horizontal Alignment	Left ▼ Left ▼ Left ▼
Title Vertical Alignment	middle ▼ middle ▼ middle ▼
TOOLBAR BUTTONS	

- **Table Width** - Defines the length for the application table. this option is available to select **Pixel** or **Percent** in the table width unit.
- **Table Width Unit** - The unit used to define the width of the application. Automatic (Width automatically defined according to the size of the fields); Pixel (Width defined by pixels, that should be informed in the Form: 800px); Percent (Width defined in percentage, and should be informed in the Form: 80%)
- **Format Row Counter** - Configuration of visualization of the display summary option of the Grid. The display summary option is displayed and executed only if the line option in the toolbar isn't enabled
- **Table a Group By** - Defines the left margin of the Group By.
- **Separates the Group** - Defines the margin between two Group By.
- **Records per page** - Available values in the Record button, on the toolbar
- **Templates** - Defines the templates used by default on the Header and Footer. It's possible to inform the values to the variables of the selected templates, clicking on the editing icon right beside the field of the template selection.
- **Grid fields alignment** - It's possible to define the default alignment specifically for the fields that contain numbers and dates and the other types of fields.
- **Toolbar Buttons** - Define the buttons that'll be set up by default in the toolbar for the new applications. We can set up the toolbar for the Grid, Detail, and Summary.

Form

Table Width	<input type="text" value="100"/>
Table Width Unit	<input type="text" value="Percent"/>
Format Row Counter	<input type="text"/>
Automatic tab	<input type="radio"/> Yes <input checked="" type="radio"/> No
Highlight Text on Focus	<input checked="" type="radio"/> Yes <input type="radio"/> No
Highlight Field with Error	<input type="radio"/> Yes <input checked="" type="radio"/> No
ERROR SETTINGS	
Error Position on the field	<input type="text" value="up"/>
Show the Error Title in the Application	<input checked="" type="radio"/> Yes <input type="radio"/> No
Show the Error Title in the Field	<input checked="" type="radio"/> Yes <input type="radio"/> No
Error Title	<input type="text"/>
REQUIRED FIELDS	
Display message	<input type="radio"/> Yes <input checked="" type="radio"/> No
TEMPLATES	
Header	<input type="text" value="Flat"/>
Footer	<input type="text" value="Default"/>
TOOLBAR BUTTONS	

- **Table Width** - Defines the length for the application table. this option is available to select **Pixel** or **Percent** in the table width unit.
- **Table Width Unit** - The unit used to define the width of the application. Automatic (Width automatically defined according to the size of the fields); Pixel (Width defined by pixels, that should be informed in the Form: 800 pixels); Percent (Width defined in percentage, and should be informed in the Form: 80%)
- **Format Row Counter** - Configuration of visualization of the display summary option of the Form on a horizontal orientation. The Display summary option is displayed and executed only if the line option in the toolbar isn't enabled
- **Automatic tab** - Enables the automatic tab when finishing to inform a field.
- **Highlight Text on Focus** - Allows the content of the fields to be Highlighted when the field is focused.
- **Highlight Field with Error** - When occurring an error on a field, it is also required, for example, the focus will be applied to this field.
- **Error Position on the field** - Position of the field error.
- **how the Error Title in the Application** - Defines the display of the title of the application error.
- **how the Error Title in the Field** - Defines the display of the error Title for the field.
- **Error Title** - Defines the error Title.
- **Display message** - Defines the display for the message if the field is required.
- **Templates** - Defines the templates used by default on the Header and Footer. It's possible to inform the values to the variables of the selected templates, clicking on the editing icon right beside the field of the template selection.
- **Toolbar Buttons** - Define the buttons that'll be set up by default in the toolbar for the new applications.

Control

Highlight Field with Error	<input type="radio"/> Yes <input checked="" type="radio"/> No
ERROR SETTINGS	
Error Position on the field	<input type="text" value="up"/>
Show the Error Title in the Application	<input checked="" type="radio"/> Yes <input type="radio"/> No
Show the Error Title in the Field	<input type="radio"/> Yes <input checked="" type="radio"/> No
Error Title	<input type="text"/>
REQUIRED FIELDS	
Display message	<input type="radio"/> Yes <input checked="" type="radio"/> No
TEMPLATES	
Header	<input type="text" value="Flat"/>
Footer	<input type="text" value="Default"/>

- **Highlight Text on Focus** - Allows the content of the fields to be highlighted when the field is focused.
- **Highlight Field with Error** - When occurring, an error on a field is also required. For example, the focus will be applied to this field.
- **Error Position on the field** - Position of the field error.
- **how the Error Title in the Application** - Defines the display of the title of the application error.
- **how the Error Title in the Field** - Defines the display of the error Title for the field.
- **Error Title** - Defines the error Title.
- **Display message** - Defines the display for the message if the field is required.
- **Templates** - Defines the templates used by default on the Header and Footer. It's possible to inform the values to the variables of the selected templates, clicking on the editing icon right beside the field of the template selection.

Search

Empty Search	<input checked="" type="radio"/> Yes <input type="radio"/> No
Search Criteria	AND
Horizontal Alignment	Center
Display Condition	<input type="radio"/> Yes <input checked="" type="radio"/> No
Keep Values	<input checked="" type="radio"/> Yes <input type="radio"/> No
Keep columns and sorting	<input checked="" type="radio"/> Yes <input type="radio"/> No
Table Width	100
Table Width Unit	Percent
Use Iframe	<input type="radio"/> Yes <input checked="" type="radio"/> No
Show Results	<input checked="" type="radio"/> Yes <input type="radio"/> No
Iframe Height	1500

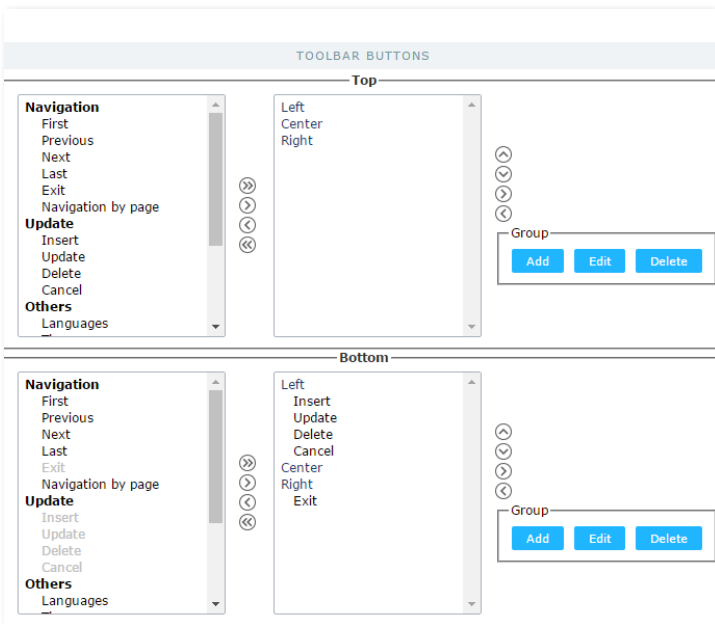
ERROR SETTINGS	
Error Position on the field	up
Show the Error Title in the Application	<input checked="" type="radio"/> Yes <input type="radio"/> No
Show the Error Title in the Field	<input type="radio"/> Yes <input checked="" type="radio"/> No
Error Title	

TEMPLATES	
Header	Flat
Footer	Default

TOOLBAR BUTTONS

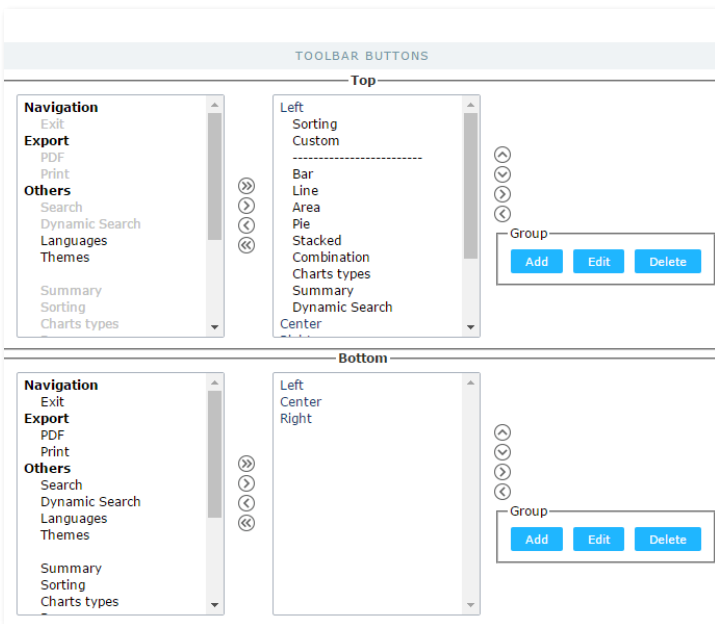
- **Empty Search** - Defines how the search treats the conditions. "AND" (Should satisfy all the conditions so that the search can return some results) "OR" (Should satisfy any condition informed so that the search can return some results)
- **horizontal Alignment** - Horizontal Alignment of the search.
- **isplay Condition** - Displays the condition of the search so that the user can make their choice. ("AND" or "OR")
- **keep Values** - Keep the values of the previous searches done when reaccessing the search.
- **keep Columns and Order Selection** - Preserve the position of the fields and the sorting of the previous search.
- **able Width** - Defines the length for the application table. this option is available to select **Pixel** or **Percent** in the table width unit.
- **able Width Unit** - The unit used to define the width of the application. Automatic (Width automatically defined according to the size of the fields); Pixel (Width defined by pixels, that should be informed in the Form: 800 pixes); Percent (Width defined in percentage, and should be informed in the Form: 80%)
- **se Iframe** - Allows the use and the display of the application searched in the same page within an iframe.
- **nitial State** - Defines how the application is displayed at first when using the search with an iframe, if the application is displayed at first or only the search is displayed.
- **frame Height** - Defines the height of the iframe in pixels.
- **rror Position on the field** - Position of the field error.
- **how the Error Title in the Application** - Defines the display of the title of the application error.
- **how the Error Title in the Field** - Defines the display of the error Title for the field.
- **rror Title** - Defines the error Title.
- **isplay message** - Defines the display for the message if the field is required.
- **emplates** - Defines the templates used be default on the Header and Footer. It's possible to inform the values to the variables of the selected templates, clicking on the editing icon right beside the field of the template selection.
- **oolbar Buttons** - Define the buttons that'll be setup by default in the toolbar for the new applications.

Calendar



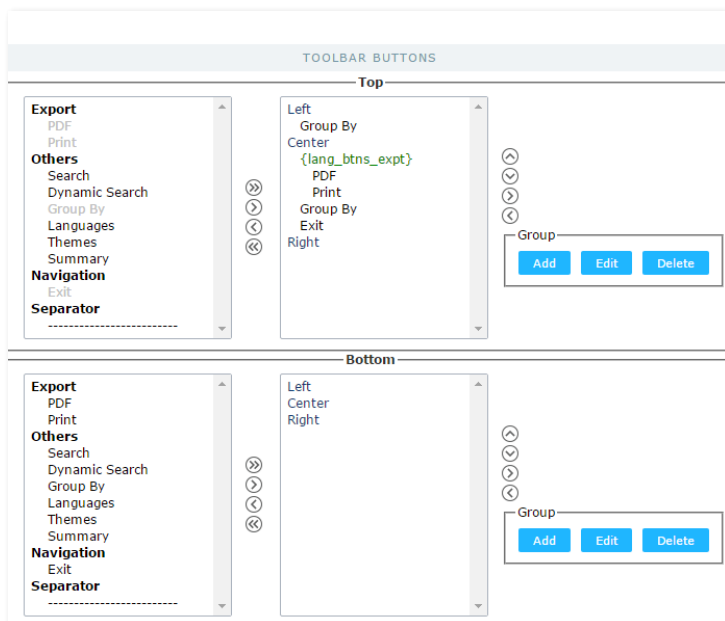
- **oolbar Buttons** - Define the default buttons in the toolbar for the new applications.

New Chart



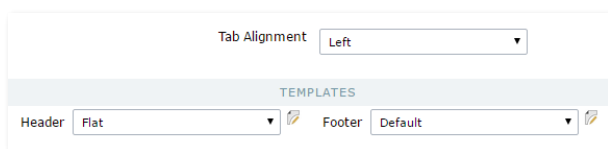
- **oolbar Buttons** - Define the buttons that will be set up by default in the toolbar for the new applications.

Old Chart



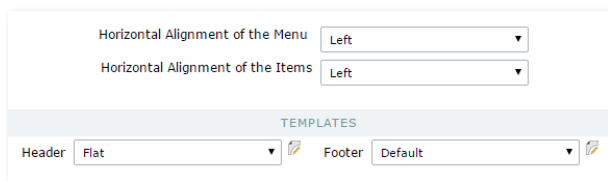
- **oolbar Buttons** - Define the default buttons in the toolbar for the new applications.

Tab



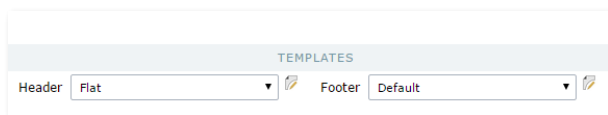
- **ab Alignment** - Display alignment of the tabs in the application.
- **emplates** - Defines the templates used by default on the Header and Footer. It's possible to inform the values to the variables of the selected templates, clicking on the editing icon right beside the field of the template selection.

Menu



- **orizontal Alignment of the Menu** - Menu alignment.
- **orizontal Alignment of the Items** - Menu item alignment
- **emplates** - Defines the templates used by default on the Header and Footer. It's possible to inform the values to the variables of the selected templates, clicking on the editing icon right beside the field of the template selection.

Dashboard



- **emplates** - Defines the templates used by default on the Header and Footer. It's possible to inform the values to the variables of the selected templates, clicking on the editing icon right beside the field of the template selection.

HTML Editor

The HTML Editor has to options of configurations, depending on the option selected on **Use template in the HTML Editor** .

- **se template in the HTML Editor** - Selecting Yes (Image Below), you use the new templated of the HTML Editor field, these themes should be set up in **Layout > HTML Editor Templates**.
- **TML Editor template** - Defines the template for editing of the field HTML Editor.
- **ield Preview** - Default Text of the HTML Editor Fields.

ATTRIBUTE VALUE

Use template in the HTML Editor Yes No

HTML Editor template

File Edit Insert View Format Table Tools

Formats B I [List of icons]

- **se template in the HTML Editor** - Selecting No (Image Below), you won't be using the new templated of the HTML Editor field. In this case, the options should be set up in the field.
- **roperties** - Defines the display properties.
- **ocation** - Defines the position of the items in the field. Top or Bottom
- **uttons Alignment** - Defines the alignment of the items in the field. Left, Center, Right.
- **tatus bar** - Position of the Status Bar.
- **oolbar Count** - Amount of the bars to organize the items.
- **uttons' Organization** - Set the available items in the bars.

ATTRIBUTE VALUE

Use template in the HTML Editor Yes No

Properties

Localization

Buttons Alignment

Status bar

Toolbar Count

Buttons' Organization

B I [List of icons] Font Family Font Size [List of icons]

Internal Libraries

Define the libraries available in the project.




Level

sc_ssn.php

- **evel** - Access level of the library.
- **ist of Libraries** - List of all the libraries available for use in the project.

Lists all the versions of the current project. The Version History page manages the versions of the project, created in the option of version incrementation, here we can return to earlier versions, open a previous version or delete created versions.

Version's description ?

Version	Description	Creation	Delete	Status	Edit
1.0.0	Prj:samples-v.1	03/21/17 09:57			

- **ersion** - Project Version.
- **escription** - Project version description.
- **reation** - Date of the version creation.
- **elete** - Delete the selected version of the project and all its applications. Applications from the other versions are not affected.
- **tatus** - Defines if the version of the project is opened or closed.
- **dit** - Allows to edit the selected version of the project. This option is available for the version that have the status Open.

Allows the control over the project in development, separating it in versions. The project version control allows you to protect the functionalities existing in the system, before doing any important modifications, by creating a new version of the current project.

The version increment allows that only the latest version of the project to be increased.

Increment Project Version samples

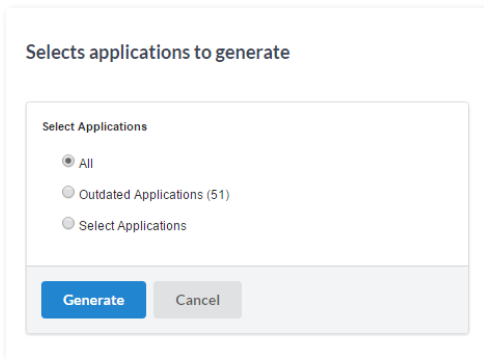
Current Version **1.0**

New Version

Description

- **urrent Version** - Informs the current version of the project.
- **ew Version** - Defines the number of the new version of the project.
- **escription** - New project description.

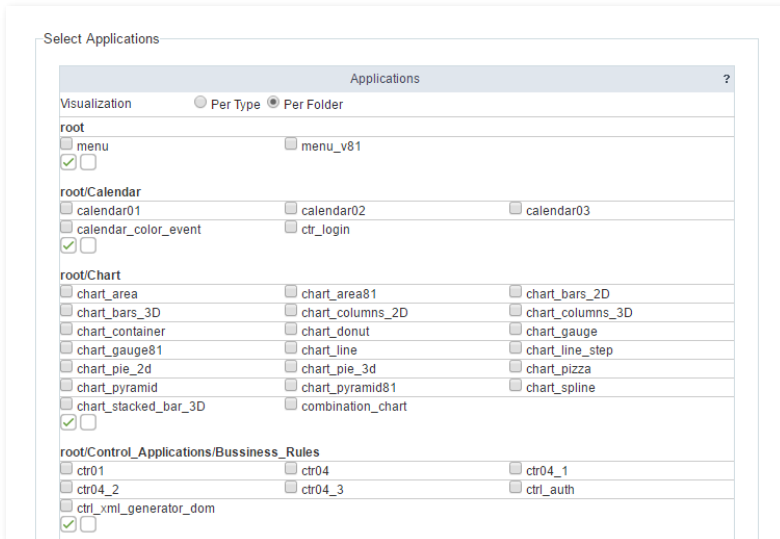
Allows you to generate the source code for all the applications in the project, select the chosen application, or generate the source code for the outdated ones.



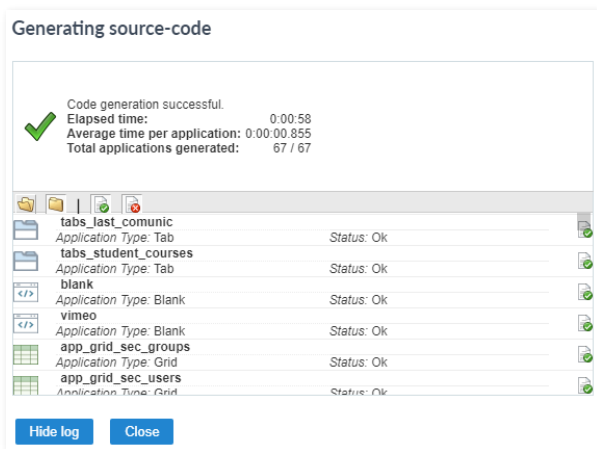
- **All** - Generates the source code for all the applications in the current project when selecting this option, you are redirected to the page where the code is generated, where you can also see the log.
- **Outdated Applications** - Generates the source code only for the outdated applications (recently modified applications that weren't generated yet), when selecting this option, you will be redirected to the page where the code will be generated, where you can see the log.
- **Select Applications** - This option allows you to select the applications that you want to generate the source code, when selecting this option, you will be redirected to the application selection page.

Selecting Applications

All the applications from the project are listed here, and you should select the applications that you want the source code to be generated. The application can be listed in two ways, using the filter to list by Application types or by folders, as shown below.



After selecting the application that will be generated, you will be redirected to the page where the code will be generated, where you can also see the log.

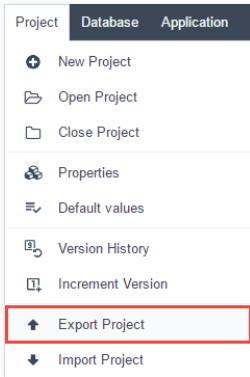


We can see in the highlighted folder of the image some filter options.

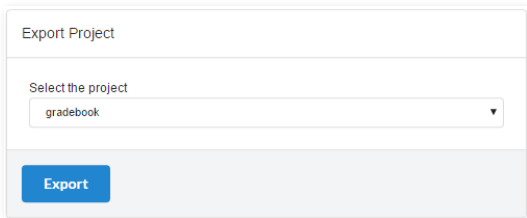
- **pen Folder Icon** - Allows possible errors to be displayed when generating the code.
- **losed Folder Icon** - Allows to hide the possible errors when generating the code.
- **ocument ok** - Displays only the applications that haven't had any problems generating the code.
- **ocument error** - Displays only the applications that has had any problems generating the code.

This option allows you to create a .zip file with all the applications from the current project and all its required files.

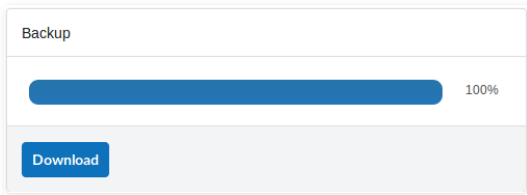
Access **File > Export Project** .



Next, select the project that you want to export.

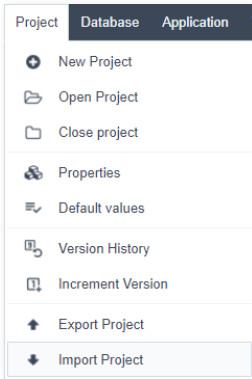


When finishing the process, the file will be available for download.

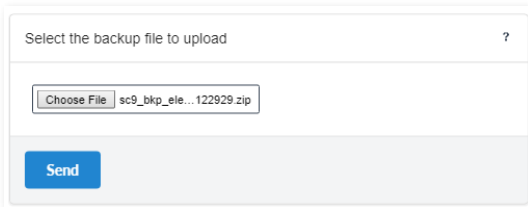


Allows you to import a project exported by ScriptCase, with all the required files.

Access **File > Import Project**.

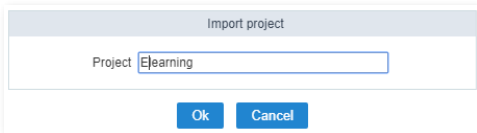


Next, we need to select the project file that we want to import.

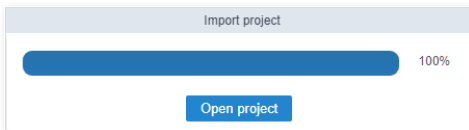


After finishing the process, we should inform the name of the project that is being imported.

In case there is a project with the same name in your ScriptCase, the project that is being imported will overwrite the existing project.

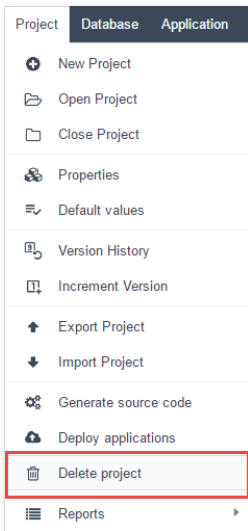


After finishing the import, we can open the project.



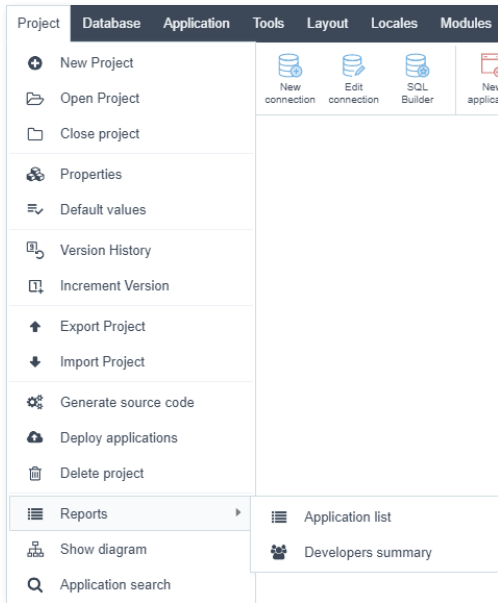
To delete a project, you need to have it opened. In the project, access **Project > Delete Project** . All the information relating to the project and the applications will be deleted.

The project exclusion process can't be undone.



We offer two types of reports for projects: Application List and Developers Summary.

To generate the reports, access **Project > Reports**



Application List

Displays a list with all the existing applications in the current project and the **SQL INSERT Queries** of the tables. To generate the report, first you need to define which information you want to be available.

Report: Applications List

Select Columns	Name Description Create user Date/Hour of Creation Date/Hour of Update Date/Hour of Generation Generation Status	⌕ ⏪ ⏩ ⏴ ⏵ ⏶ ⏷	Name Description Create user Date/Hour of Creation Date/Hour of Update Date/Hour of Generation Generation Status	⏴ ⏵ ⏶ ⏷	
Sorting	Name Description Create user Date/Hour of Creation Date/Hour of Update Date/Hour of Generation Generation Status	⌕ ⏪ ⏩ ⏴ ⏵ ⏶ ⏷	+ Name	⏴ ⏵ ⏶ ⏷	Order By: <input type="radio"/> Ascendant <input type="radio"/> Descendant
Group By	<input type="text"/>				

- **elect Columns** - Defines the columns that will be part of the report.
- **orting** - Defines the fields and the sorting that will be available.
- **rder By** - Defines the initial sorting for the report. It will use **Ascendant** by default if not informed.
- **roup By** - Defines the grouping of the report, by type (application) or by folder (Project's File Directory).

After finishing the configurations, click on **Report**.

Report: Applications List

Name	Description	Create user	Date/Hour of Creation	Date/Hour of Update	Date/Hour of Generation	Generation Status
calendar_calendar		admin	02/05/18 10:52	02/05/18 10:55	02/05/18 10:55	Generated
form_products		admin	02/07/18 16:22	02/07/18 16:24		Outdated
grid_products		admin	02/02/18 14:06	02/02/18 14:06	02/02/18 17:48	Generated
search_products		admin	02/05/18 10:27	02/05/18 10:27		Outdated

Developers Summary

Generates a Summary with every action each developer had done, showing the total applications and their types and the total of code lines that the user wrote in the applications.

Developers summary

Report Charts

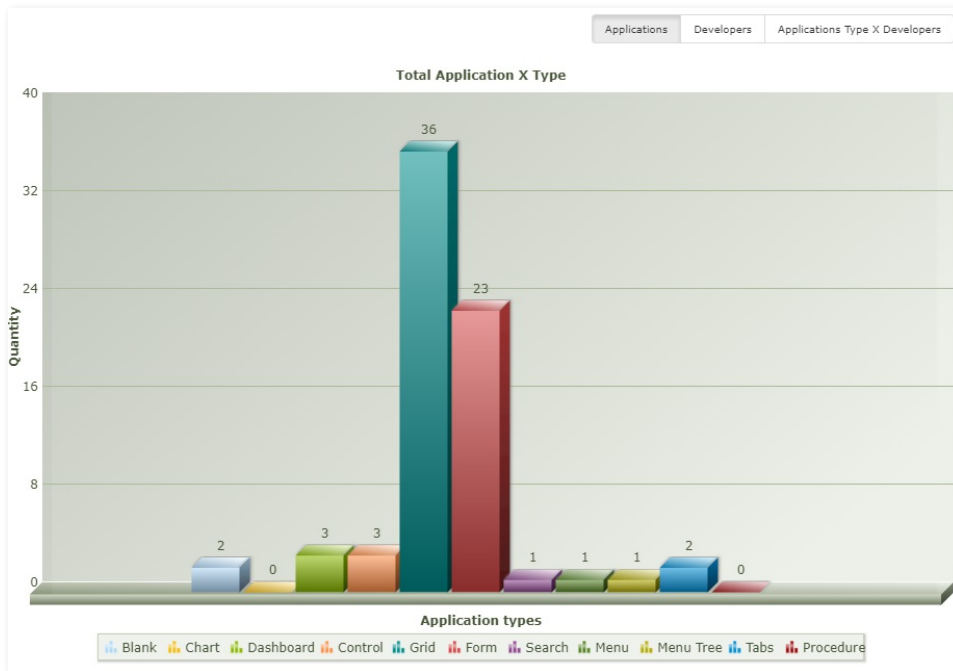
REPORT: SUMMARY BY DEVELOPER ?

Developer	Grid	Chart	Form	Control	Tabs	Procedure	Menu	Menu Tree	Search	Dashboard	Blank	Total	Code Lines
__ldap__ralves	0	0	0	0	0	0	0	0	0	0	0	0	0
admin	3	0	3	0	0	0	0	0	0	0	0	6	0
scriptcase	33	0	20	3	2	0	1	1	1	3	2	66	0
support	0	0	0	0	0	0	0	0	0	0	0	0	0

Close

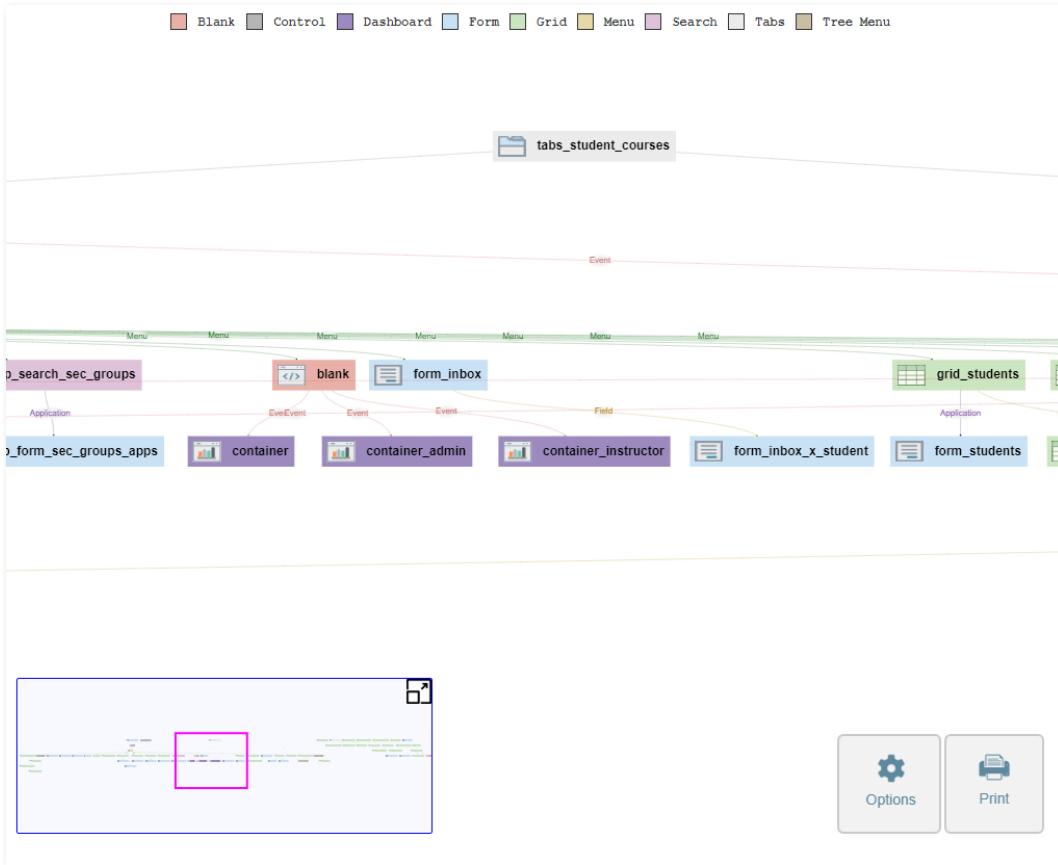
This same report is available in Charts, with three types of visualization.

- **pplications** - Total of applications in the project grouped by their type.
- **evelopers** - Total of application created by developer in the project.
- **pplications x Developers** - A junction of the other two charts.



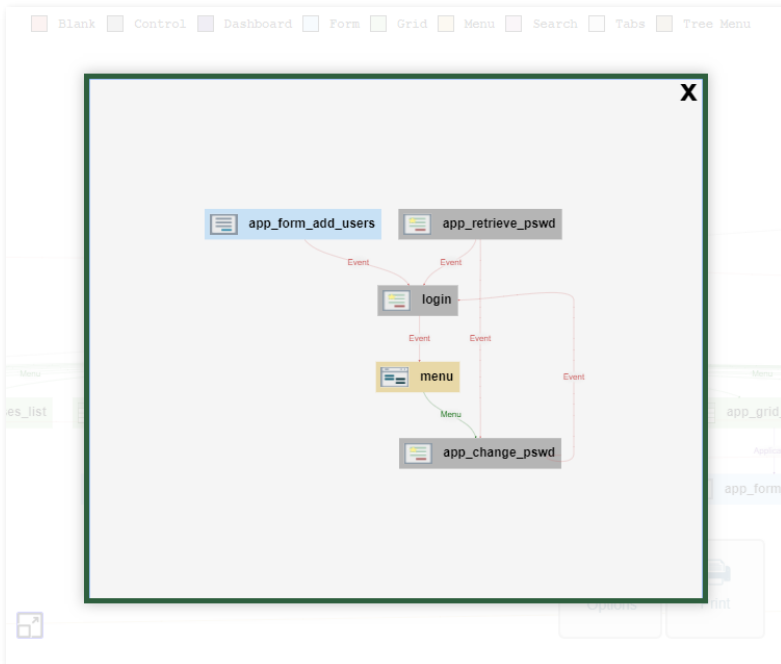
Generates a diagram of relationships between applications in the project. On this diagram, you can see in a clear way which applications are related and which is the method used to create the relationship.

On this example we're showing the relationship of all the applications in the sc_album project.

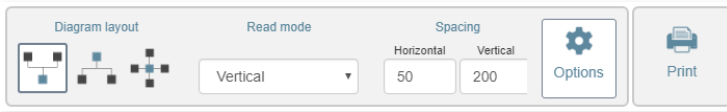


There are some options that can help visualizing the diagram.

We can display a mini map of all the diagrams, to help with the display. In order to visualize it, click on the icon on the lower left side of the diagram.



There are other options of views, instead of the mini map, as shown below.



- **Diagram Format** - Build the Diagram according to the ties positioning.
- **Reading Mode** - Defines the perspective of reading for the diagram when it's created.
- **pace** - Defines the space between the columns(Horizontally) and the layers(Vertically).

It performs a search in the project with all the content informed by the search itself.

The location on which the search will look for defined by a checkbox.

At least one of the options need to be selected.

APPLICATION SEARCH

Search word: Anything Words

All Theme
 SQL Template Name
 Field Name and Label Libraries
 Links Header and Footer
 Lookup Block Name and Label

- **All** - Performs a search in all the options listed below.
- **SQL** - Searches only in the SQL areas of the applications.
- **field Name and label** - Performs a search for the name and label of the field.
- **inks** - Performs a search only for the name of the applications.
- **lookup** - Searches inky in the lookup area of the fields.
- **themes** - Searches only for theme names.
- **template Name** - Searches only for HTML templates names.
- **libraries** - Searches for internal libraries that contain this specific name.
- **header and Footer** - Searches only for the Application Titles.
- **block Name and Label** - Searches only for the name or label of the block informed.

The results will be displayed as shown below.

Application	Item	Value
chart01 (Grid)	SQL Select Statement	SELECT orderid, customerid, employeeid, orderdate, requireddate, shippeddate, shipvia, freight, priceorder, shipcountry, shipregion, shipstate, shipcity, shipname, shipaddress, shippostalcode FROM orders
	Grid Title	(lang_header_grid_70_title)
	Detail Title	(lang_header_grid_70_title)
	Blocks	0 Name => grid_orders Text => grid_orders
chart02 (Grid)	SQL Select Statement	SELECT orderid, customerid, employeeid, orderdate, requireddate, shippeddate, shipvia, freight, priceorder, shipcountry, shipregion, shipstate,

- **application** - Name of the applications found that satisfies the search.
- **item** - Location where the search criteria was found.
- **values** - Values found that satisfies the search.

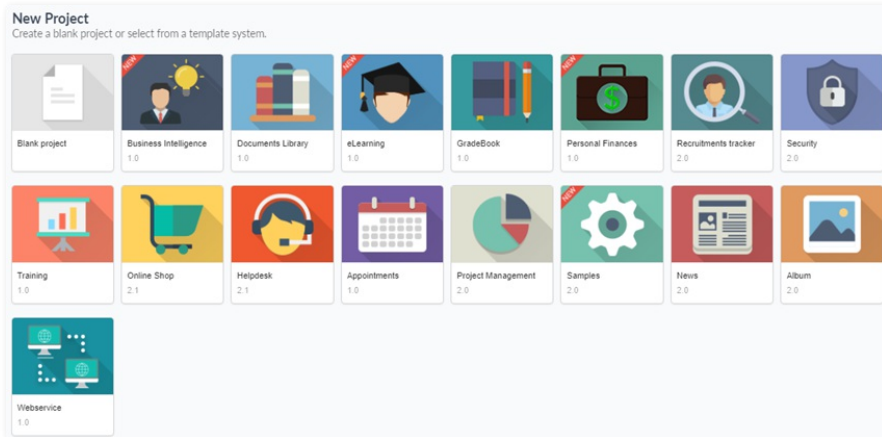
In this tutorial, you learn how to import a project template from Scriptcase. The project templates contain various samples applications for you to copy or learn how to apply Scriptcase features. You can run all the template projects to test by accessing the website [here](#).

The process of importing a template is divided into two steps, which are detailed below.

The projects created with access permission for the user, appear in [Project List](#).

Start

During the first step of the project creation, you can create a [Blank project](#) or [Import a template](#) according to the options below.



When you import a project template, Scriptcase will delivery some ready to use applications and also the database tables. Here you see the steps to import one template project, the "Samples."

First, you need to choose the project and customize all the details. Description and image are optional.

- **Project Icon** - You can select a project icon for display purposes in the project list. This field is not required, and you may change after creation inside project properties.
- **Project Name** - The project name must contain 1-32 alphanumeric characters, without space neither special characters.
- **Project Description** - Here, you give a brief description of the project, not required, also able to inform after the creation of the project.
- **Additional Project information** - Here, you can add some additional information you want for the project. This field is not required, and you may change after the creation of project properties.

Database

Our template projects are available for MySQL, Oracle, SQL Server, PostgreSQL, Access, SQLite, Firebird.

You can choose to use the default connection or to configure a new connection to your database.

Use Default Connection (SQLite)

By clicking on the option "Use default connection (SQLite)," Scriptcase undertakes to create a database, with the SQLite database, and insert all the information of tables and records. The database file with the project tables and some dummy records within the Scriptcase directory.

Creating the new project

Elearning

Step 1/4 - Creating the new project. ✓

Step 2/4 - Preparing the database and the applications to import. ✓

Step 3/4 - Importing the applications (this step may take a few minutes). ✓

Step 4/4 - Creating the database. ✓


67 Imported applications in 00:00:09


Generate source-code
Open project


The project will be created and configured with all the essential information. Generate the source code to compile and applications and then "open project" and view all the applications created or "Run" the project to open the initial application


Create A New Connection To My Database


Create a connection
Select your database system and create a connection for your project. ?



 MySQL/MariaDB



 Oracle



 MSSQL Server



 Postgres


 DB2


 Informix


 MS Access


 SQLite


 Firebird

After selecting the database you want to connect, enter the required data to perform the connection.

The information needed to create a connection varies according to the selected database. In the example below, we show with MySQL connection. For more details on how to connect your database [click here](#).

CONNECTION SECURITY FILTER ADVANCED ?

Connection Name: DBMS Driver:

Server/Host (Name or IP): Port:

Username: Password:

Database Name: List Database Create Database

Test Connection
Save

- **Connection Name**: Define the name of the connection. You can change this name in the future.
- **DBMS Driver**: Defined as MySQL PDO. It can also be MySQLi.
- **Server/Host (Name or IP)**: Enter the IP of the server where the database is. If the database is on the same machine of Scriptcase, you can use the IP **127.0.0.1** or **localhost**.
- **Port**: Define the port of the connection. By default, the port is 3306.
- **ser**: Inform the user name that you use to connect to your database.
- **assword**: Inform the password that you use to connect to your database.
- **atabase Name**: Click on the "List Database" and select the desired database.
- **est Connection**: Displays the connection status if it succeeded or not.

If you don't have a database created, click on the button "Create Database" and the window below appears.

The option of **Create Database** is available for: **MySQL/MariaDB, MSSQL Server and PostgreSQL**

Create Database

Enter a name for your database and click "OK." After that, the database created will be displayed on the database list. To create a database directly from Scriptcase, you need to have the correct permissions within the database system itself, and sometimes this permission is refused, especially when you have the database in another server. In this example, we have the database located on the same server, localhost.

Project Generation

After you finish, Scriptcase undertakes to create all the tables and insert all the information used in applications.

Creating the new project
Elearning

Step 1/4 - Creating the new project. ✓

Step 2/4 - Preparing the database and the applications to import. ✓

Step 3/4 - Importing the applications (this step may take a few minutes). ✓

Step 4/4 - Creating the database. ✓

67 Imported applications in 00:00:09

All set! Now the project will be created and configured with all the essential information. Generate the source code to compile the applications and then open project and view all the applications created.

RELATED LINKS

[Data Modeling and Project Creation](#)

[Scriptcase 9 Presentation](#)

ScriptCase Available Connections



Access

DB2

IBM DB2



Firebird



Informix



Interbase



MySQL



Oracle



PostgreSQL



SQL Server

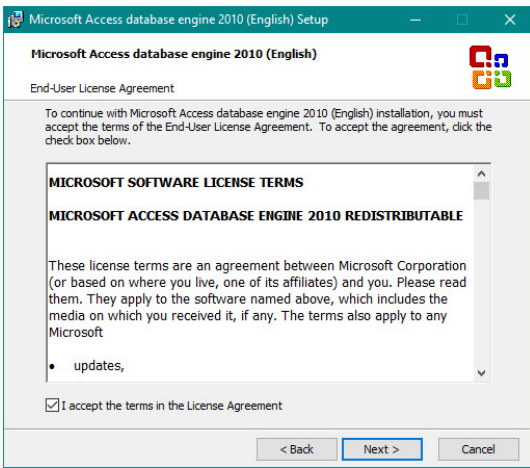


SQLite

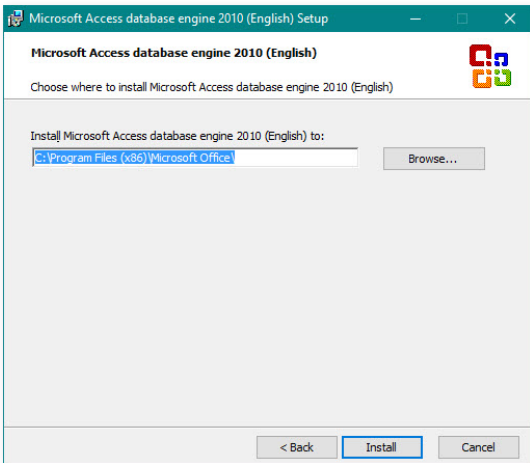


Generic ODBC

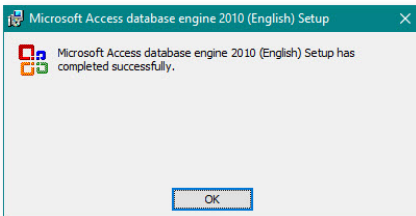




Installing the Access Database Engine.



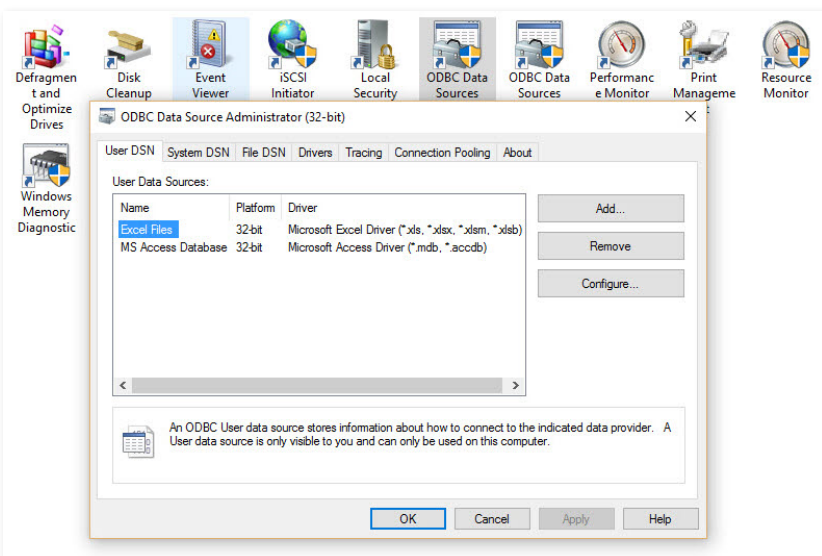
Installing the Access Database Engine.



Installing the Access Database Engine.

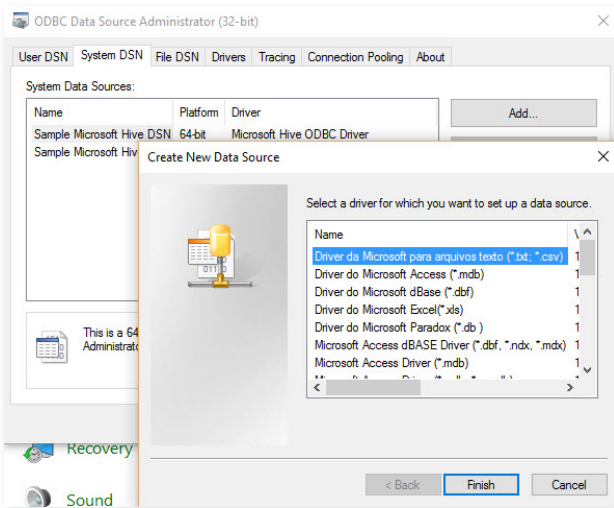
Creating A Data Source (ODBC)

- o to Control Panel> Systems and Security> Administrative Tools> ODBC Data Sources (32 or 64 bits), according to your PHP architecture.
- o f you have a data source configured, proceed to [Creating a Connection in ScriptCase](#) .



Accessing and Configuring the ODBC Data Source.

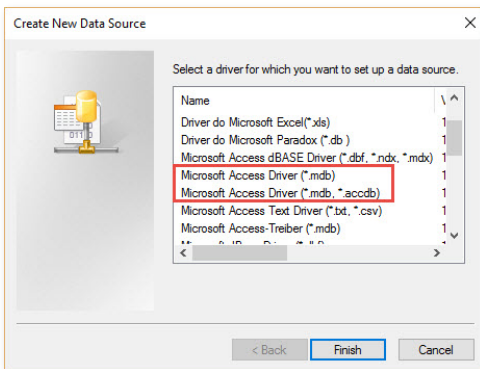
- elect the System DSN tab and click Add.



Accessing and Configuring the ODBC Data Source.

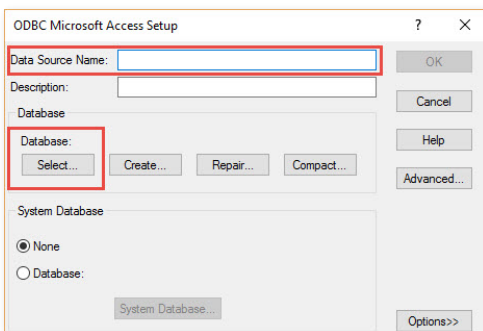
- choose one of the available drivers, you can choose from the following drivers:
 - icrosoft Access Driver (.mdb)
 - icrosoft Access Driver (.mdb, *.accdb)

Select the driver according to the file extension created by your Access database.



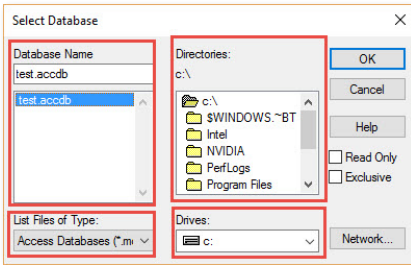
ODBC driver available.

- After selecting the driver, the following screen will be displayed:
 - Two fields are required for data source configuration:
 - **ata Source Name:** Name of the data source that will be used when creating the connection in Scriptcase.
 - **atabase:** Path pointing where the database is stored.



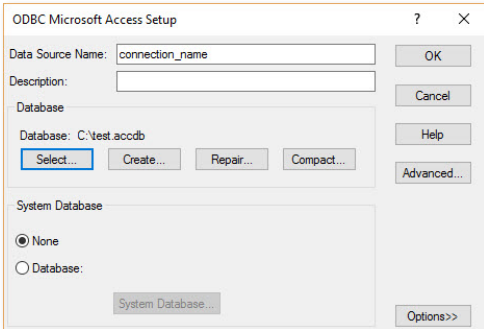
Creating the Data Source.

- After entering the Data Source Name, click "Select" to select the database that will be used.



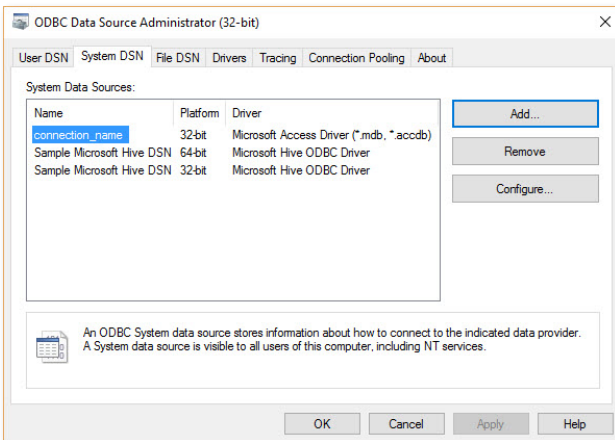
Select Database.

- **rivers:** List of computer drives (HD Partition) that will have your folders listed in Directories.
- **irectories:** Computer Drives folder list (HD Partition) selected under Drivers.
- **atabase Name:** List of files in the folder selected in Directories (Files listed according to their extensions).
- **ist Files of Type:** Files extensions listed in Database Name.
- After selecting the database, click OK to complete the configuration.



Creating the Data Source.

- Returning to the ODBC Data Source Administrator screen, you can view the already configured connections.



Creating the Data Source.

MS Access ADO

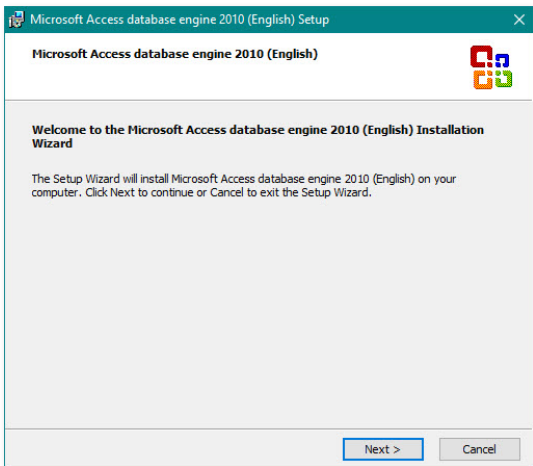
Setting Up ADO Access In Windows

Requirements

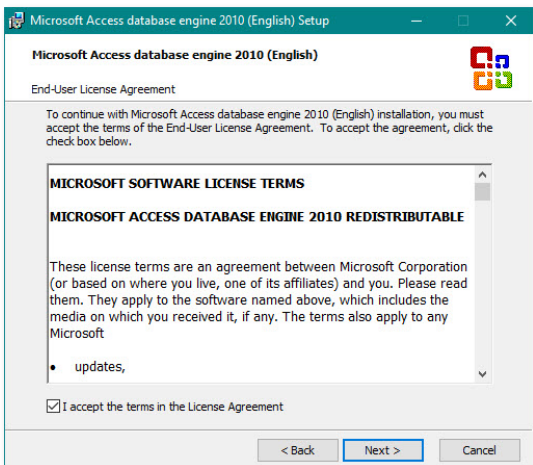
- The Access Database (.mdb or .accdb files) needs to be stored on the same server as ScriptCase.
- Have one or more tables created.
- Microsoft Microsoft 2010 Access Database Driver installed.

Installing The 2010 Access Database Engine

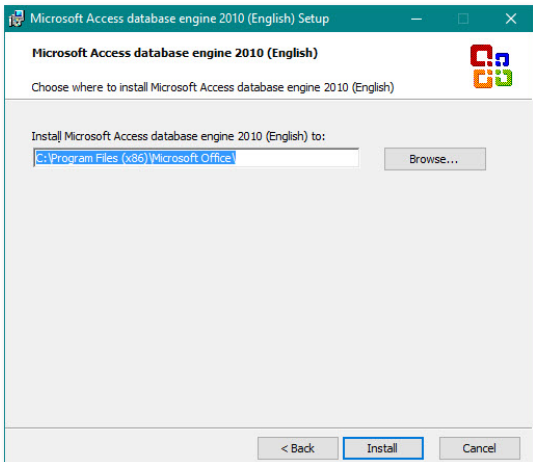
- First we must download the 2010 Access Database Engine according to the architecture of your PHP.
 - Download the 64-bit engine [click here](#)
 - Download the 32-bit engine [click here](#)
- At the end of the download, follow the Access Database Engine installation wizard.



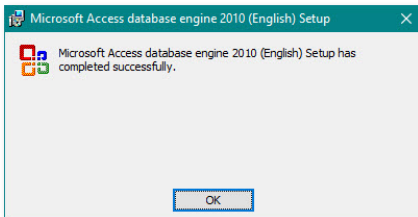
Installing the Access Database Engine.



Installing the Access Database Engine.



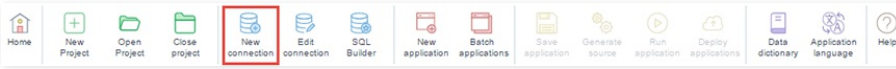
Installing the Access Database Engine.



Installing the Access Database Engine.

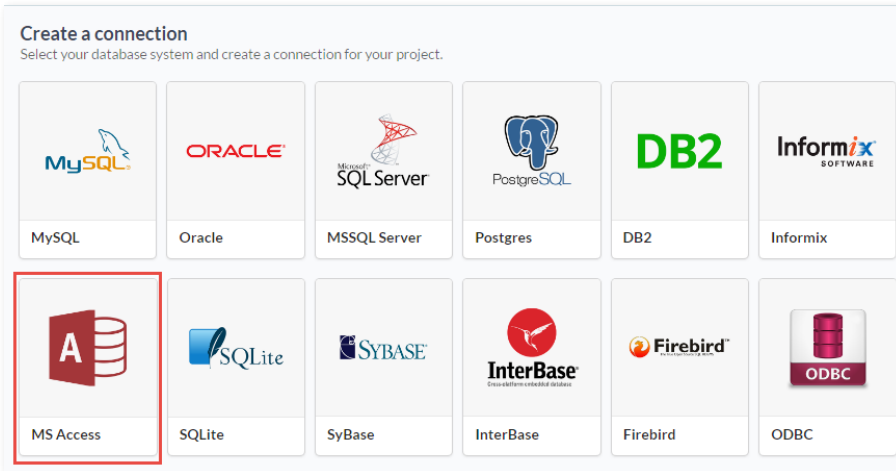
Creating a Connection with ScriptCase

- ccess ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- fter accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- After that, you will see a page with all the database connections that you can create. Select a "MS Access" connection



Selecting a Windows database connection

- **new connection** : You will place the information of the connection to your database here:

Connecting with Windows database

- **Connection Name** : Defines the name of your new connection.
 - **DBMS Driver** : Defines the driver with ODBC.
 - **ODBC Name** : On the option ODBC Name, inform the data source (ODBC) name that was created in the Windows Control Panel.
 - **Username** : Inform the username for Windows database (Just in case you are in fact using one).
 - **Password** : Inform the password for Windows database (Just in case you are in fact using one).
 - **Test Connection** : Displays the connection status (Success or Failed).
 - **Finalize**: Save connection.
- For more options on connecting, click on the **Advanced** tab:

Advanced setup for the Windows database

- **Decimal Separator** - Select the separator type, between a dot or a comma.
- **Persistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **Use the schema before table name** - Allows the use of schema before the name of the tables.
- **Filter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER ADVANCED

Show

Tables

Views

System tables

Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No ▾

Test Connection
Save

Filtering the Windows database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.



The automatic installation already comes with the IBM DB2 previously configured, only needing to install the client to make the connection.

If you're using your own environment previously configured, you'll need to enable the IBM DB2 extension in PHP.

DB2

Setting Up DB2 On Windows

Pre-Requirements

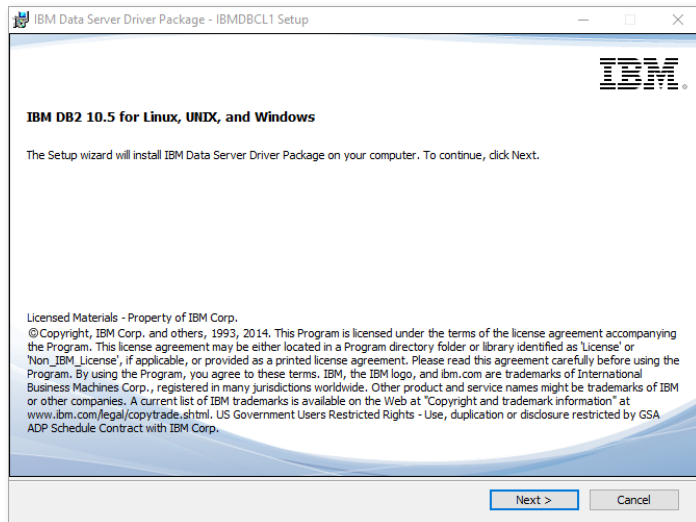
- have the database installed and setup on a server or locally.
- have at least one table created.
- have the Instant Client installed on the same machine as ScriptCase.

If all the requirements are met, the DB2 extension will be enabled, in this case you can proceed to [Creating a Connection with Scriptcase](#).

In case it is not enabled, follow the steps below to enable the DB2 extension:

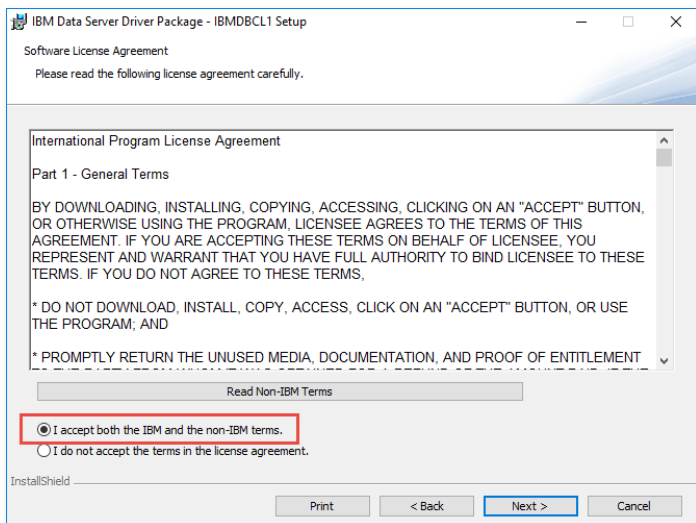
Enabling The Extension

- download the instant client compatible with the architecture of your operating system, x64 or x86 bits:
 - [DsDriver Windows \(x86 \)](#)
 - [DsDriver Windows \(x64 \)](#)
- after downloading, follow the wizard of the installation and click to advance to maintain the default values of the installer.



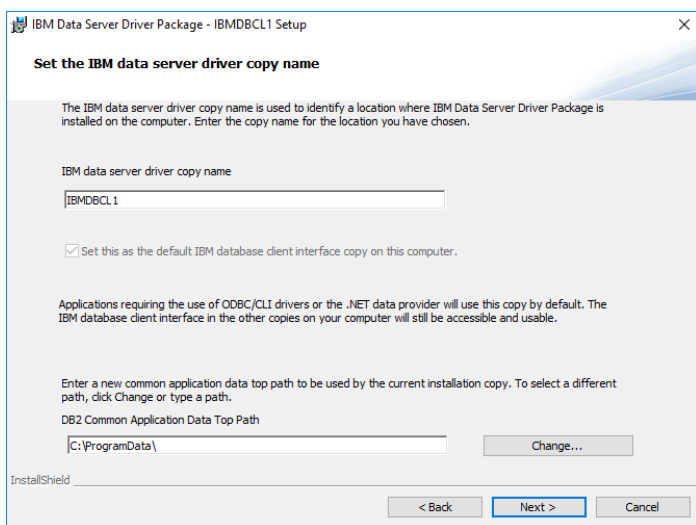
Setting up DB2 DsDriver.

- accept the terms and advance with the installation.



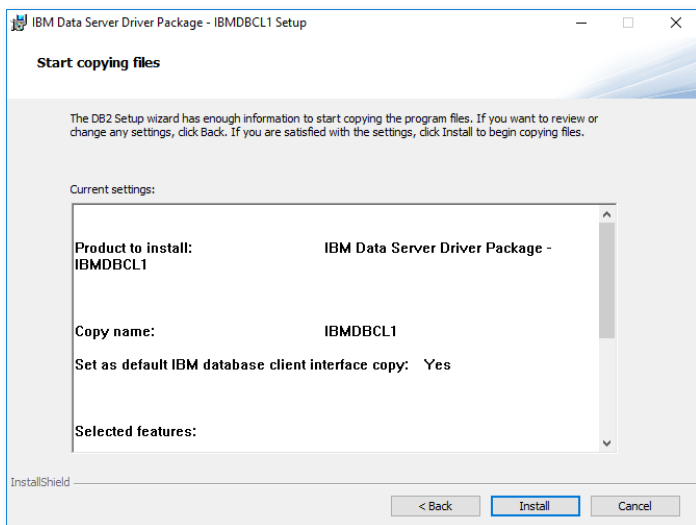
Setting up DB2 DsDriver.

- click on Advance, to maintain the default values of the installer



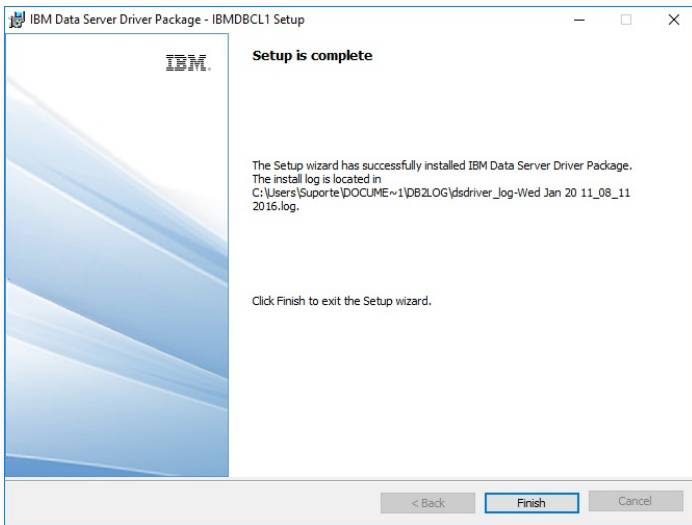
Setting up DB2 DsDriver.

- click on Install, to start the installation procedure



Setting up DB2 DsDriver.

- click on Finish, to Finish the installation.



Setting up DB2 DsDriver.

DB2 Native ODBC

Setting Up DB2 Native ODBC On Windows

Previous Requirements

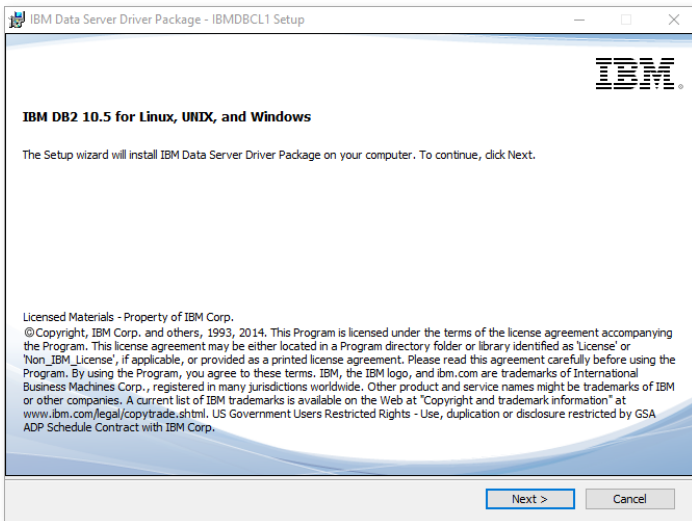
- have the database installed and setup on a server or locally.
- have at least one table created.
- have the Instant Client installed on the same machine as ScriptCase.

If all the requirements are met, the DB2 extension will be enabled, in this case you can proceed to [Creating a Connection with Scriptcase](#).

In case it is not enabled, follow the steps below to enable the DB2 extension:

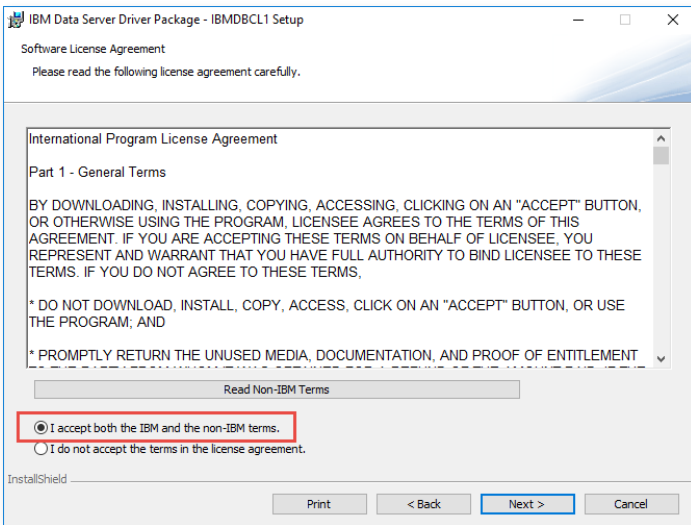
Enabling The Extension

- download the instant client compatible with the architecture of your operating system, x64 or x86 bits:
 - [DsDriver Windows \(x86 \)](#)
 - [DsDriver Windows \(x64 \)](#)
- after downloading, follow the wizard of the installation and click to advance to maintain the default values of the installer.



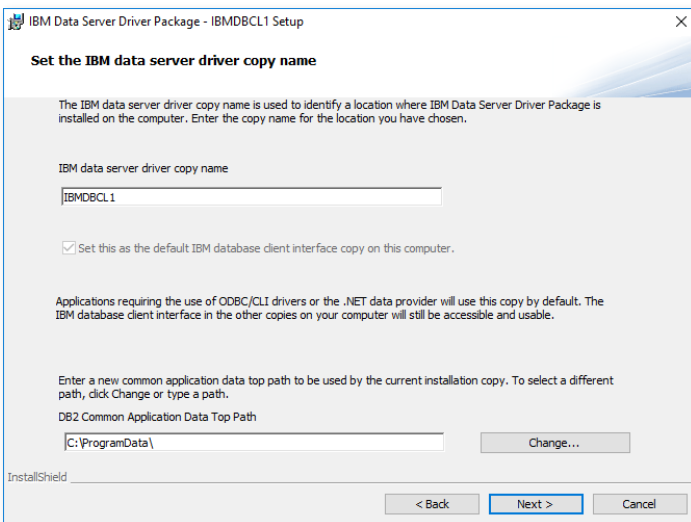
Setting up DB2 DsDriver.

- accept the terms and advance with the installation.



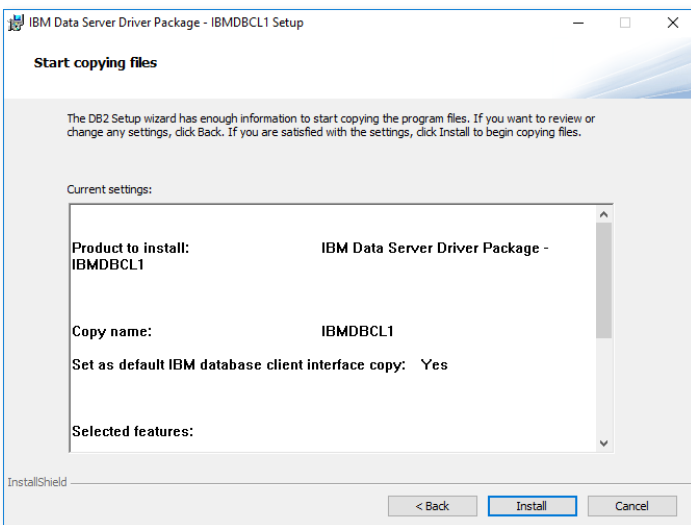
Setting up DB2 DsDriver.

- click on Advance, to maintain the default values of the installer



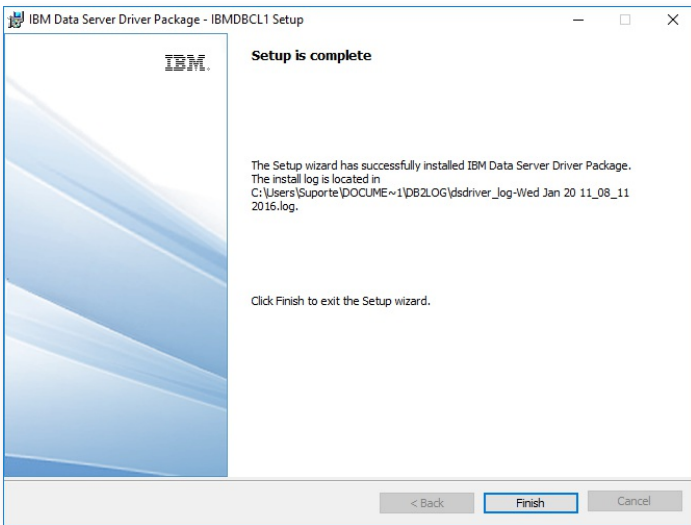
Setting up DB2 DsDriver.

- click on Install, to start the installation procedure



Setting up DB2 DsDriver.

- click on Finish, to Finish the installation.



Setting up DB2 DsDriver.

DB2 Generic ODBC

Setting Up DB2 Generic ODBC On Windows

Previous Requirements

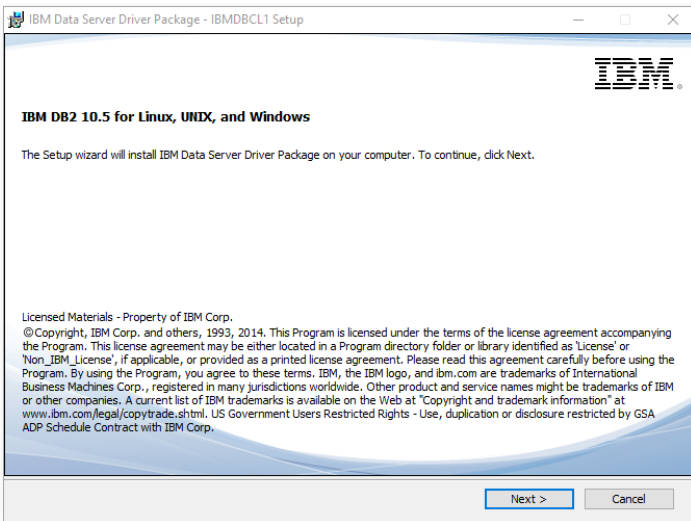
- have the database installed and setup on a server or locally.
- have at least one table created.
- have the Instant Client installed on the same machine as ScriptCase.

If all the requirements are met, the DB2 extension will be enabled, in this case you can proceed to [Creating a Connection with Scriptcase](#).

In case it is not enabled, follow the steps below to enable the DB2 extension:

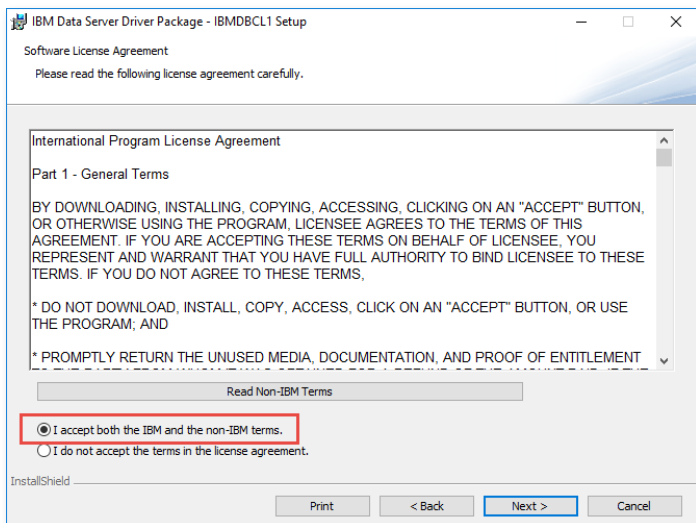
Enabling The Extension

- download the instant client compatible with the architecture of your operating system, x64 or x86 bits:
 - [DsDriver Windows \(x86 \)](#)
 - [DsDriver Windows \(x64 \)](#)
- after downloading, follow the wizard of the installation and click to advance to maintain the default values of the installer.



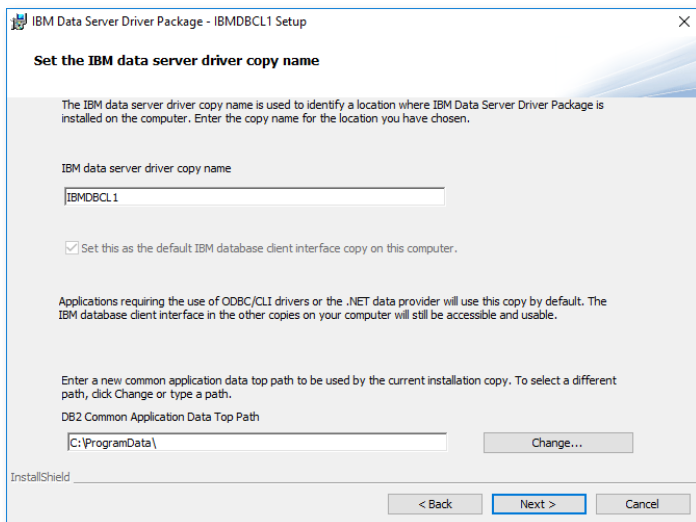
Setting up DB2 DsDriver.

- accept the terms and advance with the installation.



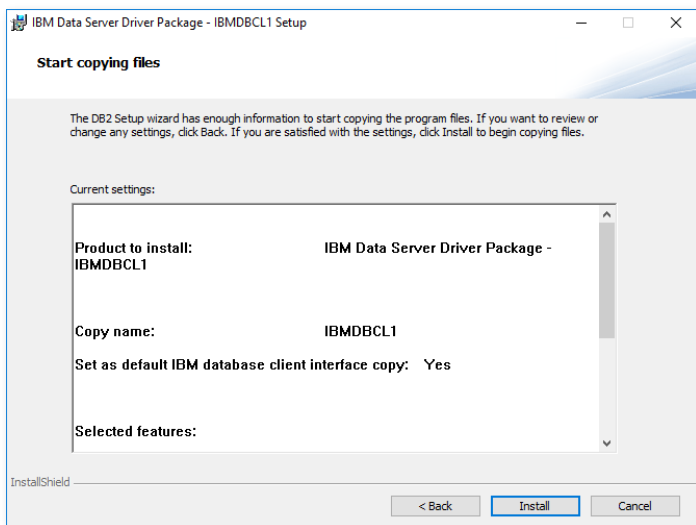
Setting up DB2 DsDriver.

- click on Advance, to maintain the default values of the installer



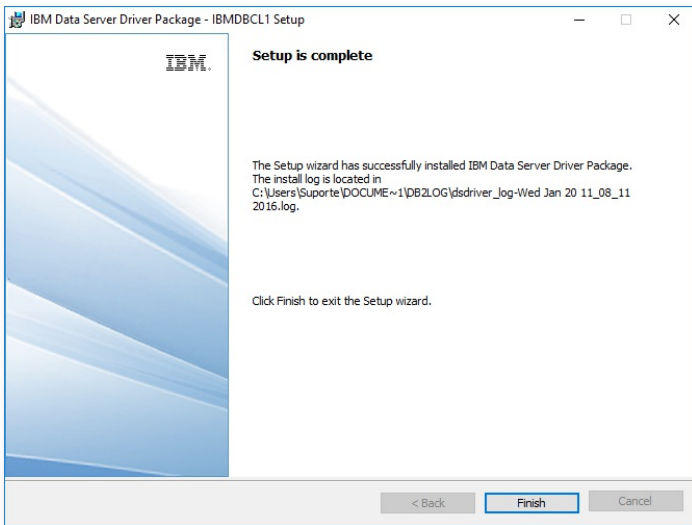
Setting up DB2 DsDriver.

- click on Install, to start the installation procedure



Setting up DB2 DsDriver.

- click on Finish, to Finish the installation.



Setting up DB2 DsDriver.

DB2 Generic ODBC 6 or Lower

Setting Up DB2 Generic ODBC 6 Or Lower On Windows

Previous Requirements

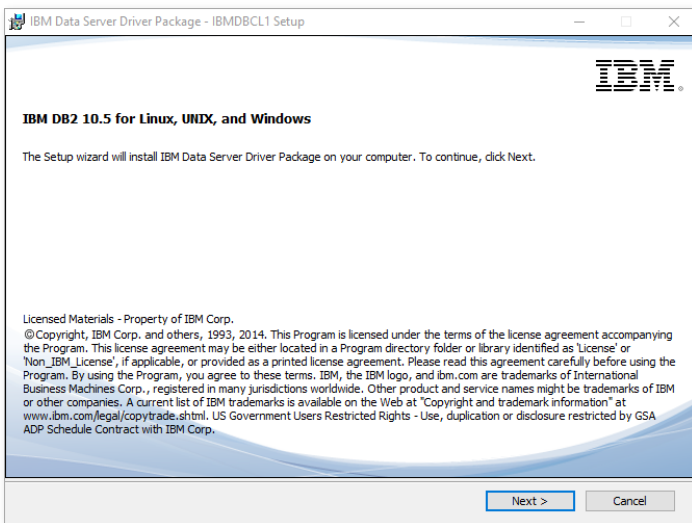
- have the database installed and setup on a server or locally.
- have at least one table created.
- have the Instant Client installed on the same machine as ScriptCase.

If all the requirements are met, the DB2 extension will be enabled, in this case you can proceed to [Creating a Connection with Scriptcase](#).

In case it is not enabled, follow the steps below to enable the DB2 extension:

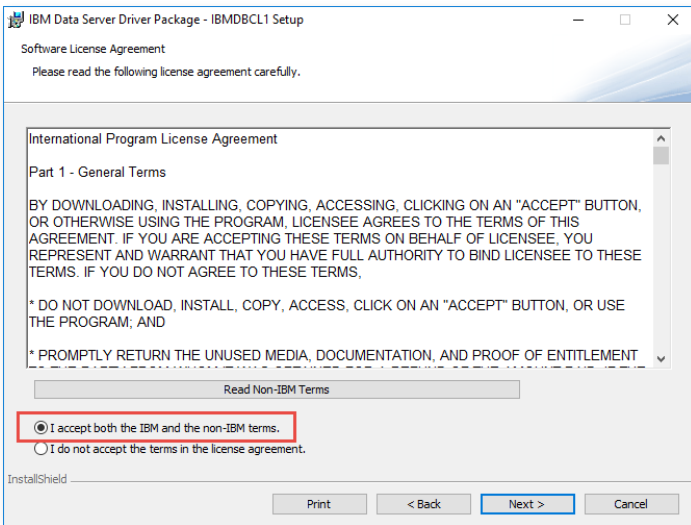
Enabling The Extension

- download the instant client compatible with the architecture of your operating system, x64 or x86 bits:
 - [DsDriver Windows \(x86 \)](#)
 - [DsDriver Windows \(x64 \)](#)
- after downloading, follow the wizard of the installation and click to advance to maintain the default values of the installer.



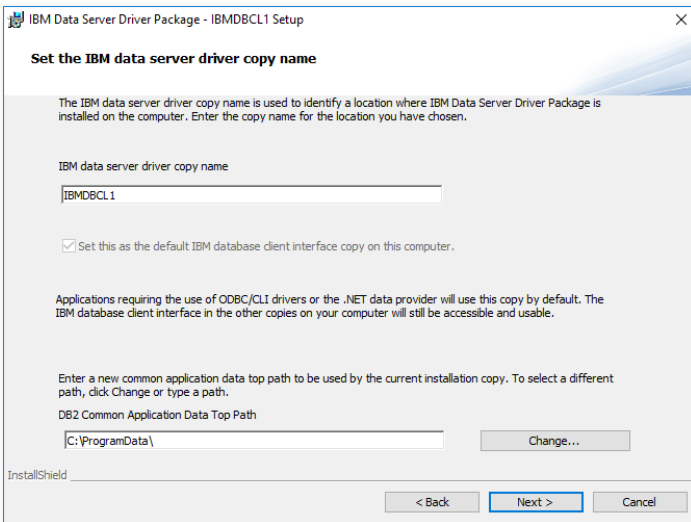
Setting up DB2 DsDriver.

- accept the terms and advance with the installation.



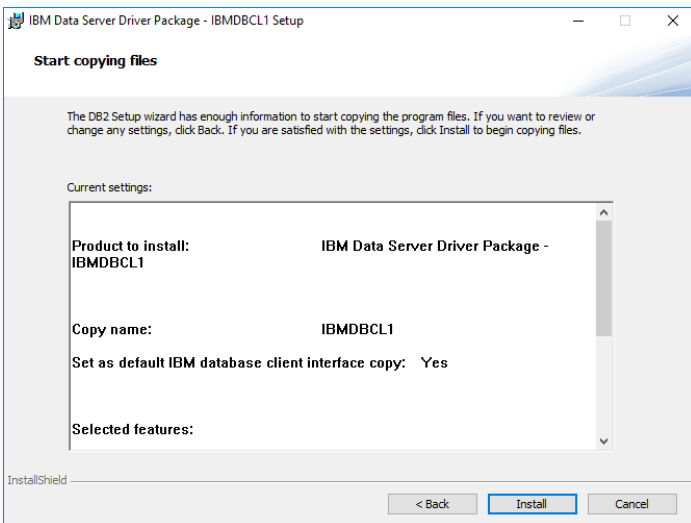
Setting up DB2 DsDriver.

- click on Advance, to maintain the default values of the installer



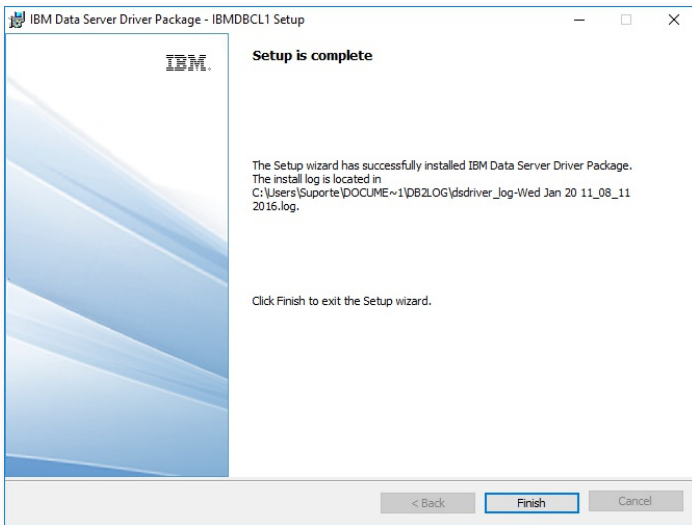
Setting up DB2 DsDriver.

- click on Install, to start the installation procedure



Setting up DB2 DsDriver.

- click on Finish, to Finish the installation.



Setting up DB2 DsDriver.

DB2 PDO

Setting Up DB2 PDO On Windows

Previous Requirements

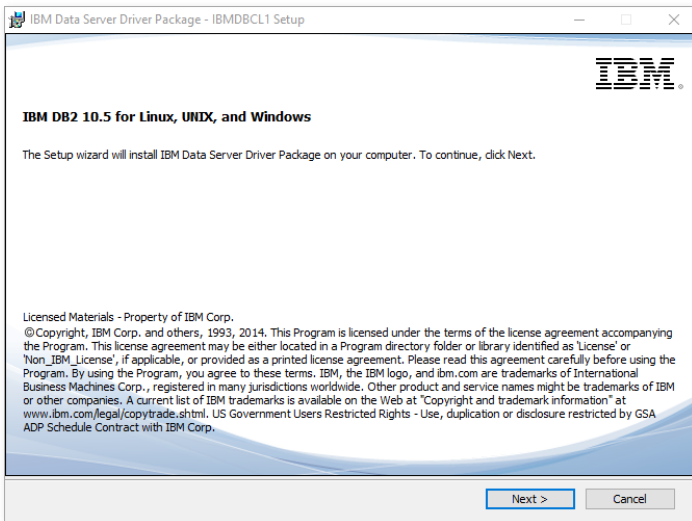
- have the database installed and setup on a server or locally.
- have at least one table created.
- have the Instant Client installed on the same machine as ScriptCase.

If all the requirements are met, the DB2 extension will be enabled, in this case you can proceed to [Creating a Connection with Scriptcase](#).

In case it is not enabled, follow the steps below to enable the DB2 extension:

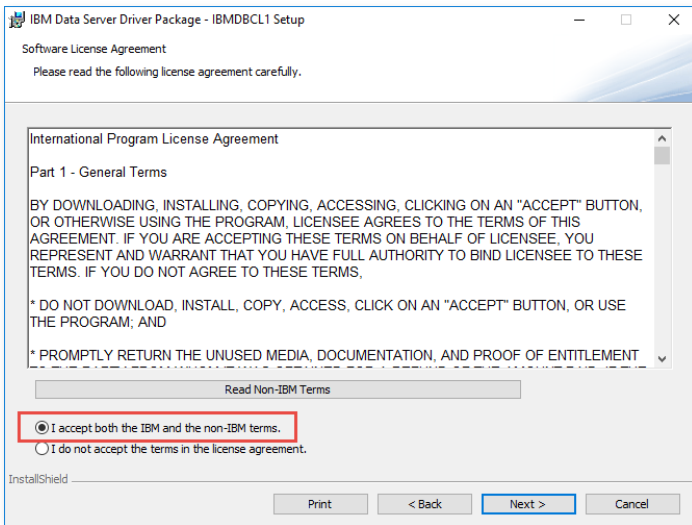
Enabling The Extension

- download the instant client compatible with the architecture of your operating system, x64 or x86 bits:
 - [DsDriver Windows \(x86 \)](#)
 - [DsDriver Windows \(x64 \)](#)
- after downloading, follow the wizard of the installation and click to advance to maintain the default values of the installer.



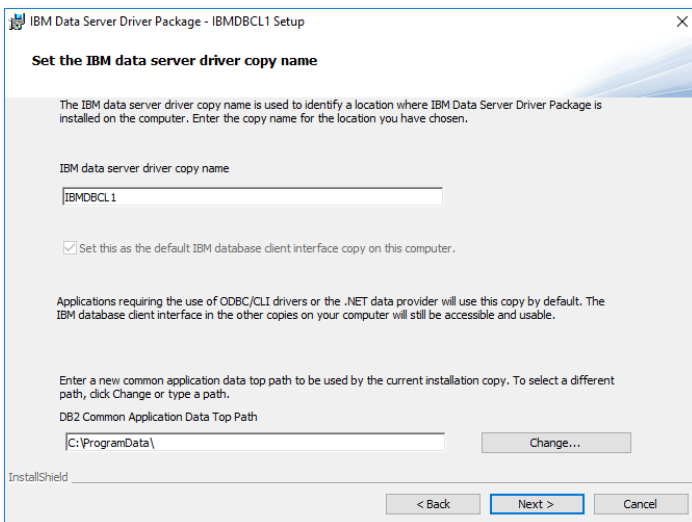
Setting up DB2 DsDriver.

- accept the terms and advance with the installation.



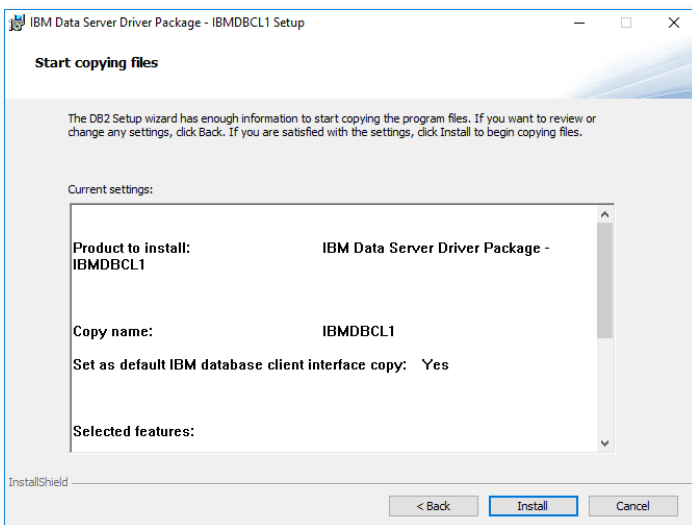
Setting up DB2 DsDriver.

- click on Advance, to maintain the default values of the installer



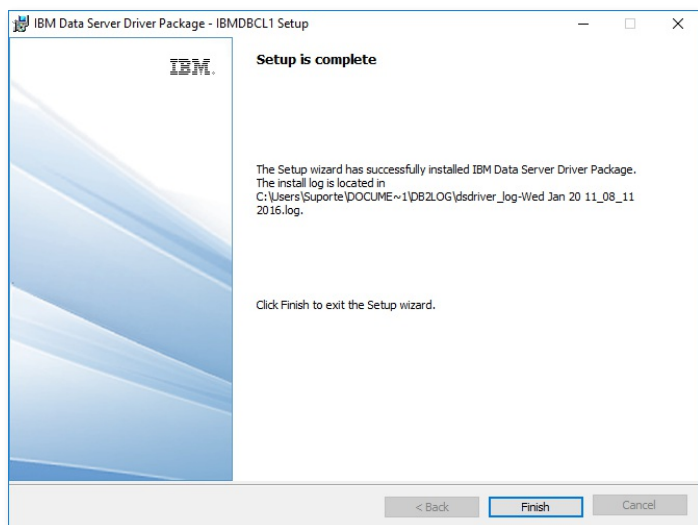
Setting up DB2 DsDriver.

- click on Install, to start the installation procedure



Setting up DB2 DsDriver.

- click on Finish, to Finish the installation.



Setting up DB2 DsDriver.

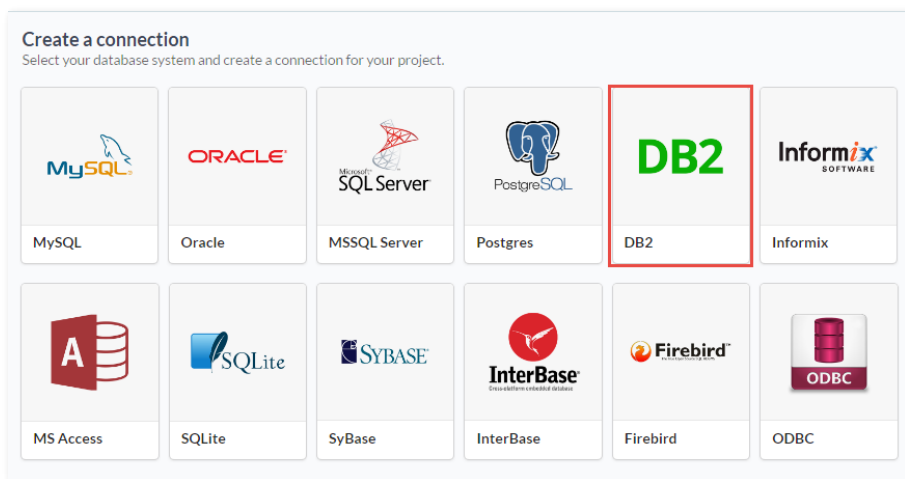
Creating a Connection with ScriptCase

- ccess ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- fter accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- fter that, you will see a page with all the database connections that you can create. Select a "DB2" connection



Selecting a Windows database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection DB2

Enter the connection details for your database DB2 ← Choose another connection

CONNECTION FILTER ADVANCED ?

Connection Name: DBMS Driver:

Server/Host (Name or IP): Port: Database Name: Schema:

Username: Password:

Note: Access the "Filter" tab to define the schema that will be loaded for this connection. Learn more by accessing this [TUTORIAL](#)

Connection Success

[Test Connection](#)

Connecting with Windows database

- **Connection Name** : Defines the name of your new connection.
- **DBMS Driver** : Allows you to select the drivers: DB2 Generic ODBC, DB2 Generic ODBC 6 or lower, DB2, DB2 Native ODBC.
- **Server/Host (Name or IP)** : Insert the name or IP of the server where the Windows database is located.
- **Database Name** : Inform the database name.
- **Schema** : You can inform the Schema (It is optional), in case you want to use the tables that are part of a Schema.
- **Username** : Inform the user to access the Windows database.
- **Password** : Inform the password to access the Windows database.
- **Test Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:

CONNECTION FILTER **ADVANCED** ?

Caution, the settings below are specific to the native ibm_db2 drive.
Only change if you know what you are doing. For more details see:
http://br2.php.net/manual/pt_BR/function.db2-connect.php

autocommit

i5_lib

i5_naming

i5_commit

i5_query_optimize

Decimal Separator

Persistent Connection

Use the schema before table name

Advanced setup for the Windows database

- **autocommit** - Enables or disables the auto commit functionality.
- **i5_lib** - A value of characters that indicates the default library that will be used to solve unqualified file references. This is not valid if the the connection is using the system nomination mode.
- **i5_naming** - Enables or disables the system nomination mode of the DB2 UDB CLI iSeries. The files are qualified when using dash (/) delimiter. Unqualified files are solved using the list of libraries to work.
- **i5_commit** - Select the desired option treating the date of the IBM DB2.
- **Decimal Separator** - Select the separator type, between a dot or a comma.
- **Persistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **Use the schema before table name** - Allows the use of schema before the name of the tables.
- **Filter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER ADVANCED

Show

Tables

Views

System tables

Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No ▼

Test Connection
Save

Filtering the Windows database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.

The automatic installation already comes with the IBM DB2 previously configured, only needing to install the client to make the connection.

If you're using your own environment previously configured, you'll need to enable the IBM DB2 extension in PHP.

DB2

Setting Up DB2 On Linux

The automatic installation already comes with the IBM DB2 previously configured, only needing to install the client so that you can do the connection.

- install the dependencies below:

UBUNTU\DEBIAN

```
- sudo apt-get update
- sudo apt-get install libaio1
- sudo apt-get install libncurses5
- sudo apt-get install alien
- sudo apt-get install gcc-multilib g++-multilib
- sudo apt-get install libpam0g
- sudo apt-get install ksh
- sudo apt-get install unixodbc-dev unixodbc
```

CENTOS\RHEL

```
- sudo yum update
- sudo yum install libaio-devel
- sudo yum install libaio
- sudo yum install glibc
- sudo yum install compat-libstdc++-33
- sudo yum install glibc-devel
- sudo yum install libstdc++
- sudo yum install libstdc++
- sudo yum install pam
- sudo yum install ncurses-devel
- sudo yum install unixODBC
- sudo wget ftp://195.220.108.108/linux/centos/6.7/os/i386/Packages/ksh-20120801-28.el6.i686.rpm
- sudo yum install ksh-20120801-28.el6.i686.rpm
- sudo yum install nano
```

- download the instant client compatible with the architecture of your operating system, x64 or x86 bits.
 - [DsDriver Linux \(x86 \)](#)
 - [DsDriver Linux \(x64 \)](#)

Installing DsDriver IBM DB2

Architecture x86

```
sudo mkdir -p /opt/IBM/Db2
sudo cp sudo cp x86_v10.5fp8_linuxia32_dsdriver.tar.gz /opt/IBM/Db2
sudo tar -zxf /opt/IBM/Db2/x86_v10.5fp8_linuxia32_dsdriver.tar.gz
sudo ksh /opt/IBM/Db2/dsdriver/installDSDriver
```

Architecture x64

```
sudo mkdir -p /opt/IBM/x64/db2
sudo cp x64_v10.5fp8_linuxx64_dsdriver.tar.gz /opt/IBM/x64/db2
sudo tar -zxf /opt/IBM/x64/db2/x64_v10.5fp8_linuxx64_dsdriver.tar.gz
sudo ksh /opt/IBM/x64/db2/dsdriver/installDSDriver
sudo source /opt/IBM/x64/db2/dsdriver/db2profile
```

Restart the Apache Server: `sudo /etc/init.d/apachesc9 restart`

DB2 Native ODBC

Setting Up DB2 Native ODBC On Linux

The automatic installation already comes with the IBM DB2 previously configured, only needing to install the client so that you can do the connection.

- install the dependencies below:

UBUNTU\DEBIAN

```
- sudo apt-get update
- sudo apt-get install libaio1
- sudo apt-get install libncurses5
- sudo apt-get install alien
- sudo apt-get install gcc-multilib g++-multilib
- sudo apt-get install libpam0g
- sudo apt-get install ksh
- sudo apt-get install unixodbc-dev unixodbc
```

CENTOS\RHEL

```
- sudo yum update
- sudo yum install libaio-devel
- sudo yum install libaio
- sudo yum install glibc
- sudo yum install compat-libstdc++-33
- sudo yum install glibc-devel
- sudo yum install libstdc++
- sudo yum install libstdc++
- sudo yum install pam
- sudo yum install ncurses-devel
- sudo yum install unixODBC
- sudo wget ftp://195.220.108.108/linux/centos/6.7/os/i386/Packages/ksh-20120801-28.el6.i686.rpm
- sudo yum install ksh-20120801-28.el6.i686.rpm
```

UBUNTU\DEBIAN**CENTOS\RHEL**

- install nano
- download the instant client compatible with the architecture of your operating system, x64 or x86 bits.
 - [DsDriver Linux \(x86 \)](#)
 - [DsDriver Linux \(x64 \)](#)

Installing DsDriver IBM DB2**Architecture x86**

```
sudo mkdir -p /opt/IBM/Db2
sudo cp sudo cp x86_v10.5fp8_linuxia32_dsdriver.tar.gz /opt/IBM/Db2
sudo tar -zxf /opt/IBM/Db2/x86_v10.5fp8_linuxia32_dsdriver.tar.gz
sudo ksh /opt/IBM/Db2/dsdriver/installDSDriver
```

Architecture x64

```
sudo mkdir -p /opt/IBM/x64/db2
sudo cp x64_v10.5fp8_linuxx64_dsdriver.tar.gz /opt/IBM/x64/db2
sudo tar -zxf /opt/IBM/x64/db2/x64_v10.5fp8_linuxx64_dsdriver.tar.gz
sudo ksh /opt/IBM/x64/db2/dsdriver/installDSDriver
sudo source /opt/IBM/x64/db2/dsdriver/db2profile
```

Restart the Apache Server: `sudo /etc/init.d/apachesc9 restart`

DB2 Generic ODBC**Setting Up DB2 Generic ODBC On Linux**

The automatic installation already comes with the IBM DB2 previously configured, only needing to install the client so that you can do the connection.

- install the dependencies below:

UBUNTU\DEBIAN

```
- sudo apt-get update
- sudo apt-get install libaio1
- sudo apt-get install libncurses5
- sudo apt-get install alien
- sudo apt-get install gcc-multilib g++-multilib
- sudo apt-get install libpam0g
- sudo apt-get install ksh
- sudo apt-get install unixodbc-dev unixodbc
```

CENTOS\RHEL

```
- sudo yum update
- sudo yum install libaio-devel
- sudo yum install libaio
- sudo yum install glibc
- sudo yum install compat-libstdc++-33
- sudo yum install glibc-devel
- sudo yum install libstdc++
- sudo yum install libstdc++
- sudo yum install pam
- sudo yum install ncurses-devel
- sudo yum install unixODBC
- sudo wget ftp://195.220.108.108/linux/centos/6.7/os/i386/Packages/ksh-20120801-28.el6.i686.rpm
- sudo yum install ksh-20120801-28.el6.i686.rpm
- sudo yum install nano
```

- download the instant client compatible with the architecture of your operating system, x64 or x86 bits
 - [DsDriver Linux \(x86 \)](#)
 - [DsDriver Linux \(x64 \)](#)

Installing DsDriver IBM DB2**Architecture x86**

```
sudo mkdir -p /opt/IBM/Db2
sudo cp sudo cp x86_v10.5fp8_linuxia32_dsdriver.tar.gz /opt/IBM/Db2
sudo tar -zxf /opt/IBM/Db2/x86_v10.5fp8_linuxia32_dsdriver.tar.gz
sudo ksh /opt/IBM/Db2/dsdriver/installDSDriver
```

Architecture x64

```
sudo mkdir -p /opt/IBM/x64/db2
sudo cp x64_v10.5fp8_linuxx64_dsdriver.tar.gz /opt/IBM/x64/db2
sudo tar -zxf /opt/IBM/x64/db2/x64_v10.5fp8_linuxx64_dsdriver.tar.gz
sudo ksh /opt/IBM/x64/db2/dsdriver/installDSDriver
sudo source /opt/IBM/x64/db2/dsdriver/db2profile
```

Restart the Apache Server: `sudo /etc/init.d/apachesc9 restart`

DB2 Generic ODBC 6 or Lower**Setting Up DB2 Generic ODBC 6 Or Lower On Linux**

The automatic installation already comes with the IBM DB2 previously configured, only needing to install the client so that you can do the connection.

- install the dependencies below:

UBUNTU\DEBIAN

```
- sudo apt-get update
- sudo apt-get install libaio1
- sudo apt-get install libncurses5
- sudo apt-get install alien
- sudo apt-get install gcc-multilib g++-multilib
- sudo apt-get install libpam0g
```

CENTOS\RHEL

```
- sudo yum update
- sudo yum install libaio-devel
- sudo yum install libaio
- sudo yum install glibc
- sudo yum install compat-libstdc++-33
- sudo yum install glibc-devel
```

UBUNTU/DEBIAN

```
sudo apt-get install ksh
- sudo apt-get install unixodbc-dev unixodbc
```

CENTOS/RHEL

```
sudo yum install libstdc++
- sudo yum install libstdc++
- sudo yum install pam
- sudo yum install ncurses-devel
- sudo yum install unixODBC
- sudo wget ftp://195.220.108.108/linux/centos/6.7/os/i386/Packages/ksh-20120801-28.el6.i686.rpm
- sudo yum install ksh-20120801-28.el6.i686.rpm
- sudo yum install nano
```

- Download the instant client compatible with the architecture of your operating system, x64 or x86 bits
 - [DsDriver Linux \(x86 \)](#)
 - [DsDriver Linux \(x64 \)](#)

Installing DsDriver IBM DB2**Architecture x86**

```
sudo mkdir -p /opt/IBM/Db2
sudo cp sudo cp x86_v10.5fp8_linuxia32_dsdriver.tar.gz /opt/IBM/Db2
sudo tar -zxf /opt/IBM/Db2/x86_v10.5fp8_linuxia32_dsdriver.tar.gz
sudo ksh /opt/IBM/Db2/dsdriver/installDSDriver
```

Architecture x64

```
sudo mkdir -p /opt/IBM/x64/db2
sudo cp x64_v10.5fp8_linuxx64_dsdriver.tar.gz /opt/IBM/x64/db2
sudo tar -zxf /opt/IBM/x64/db2/x64_v10.5fp8_linuxx64_dsdriver.tar.gz
sudo ksh /opt/IBM/x64/db2/dsdriver/installDSDriver
sudo source /opt/IBM/x64/db2/dsdriver/db2profile
```

Restart the Apache Server: `sudo /etc/init.d/apachesc9 restart`

DB2 PDO**Setting Up DB2 PDO On Linux**

The automatic installation already comes with the IBM DB2 previously configured, only needing to install the client so that you can do the connection.

- Install the dependencies below:

UBUNTU/DEBIAN

```
- sudo apt-get update
- sudo apt-get install libaio1
- sudo apt-get install libncurses5
- sudo apt-get install alien
- sudo apt-get install gcc-multilib g++-multilib
- sudo apt-get install libpam0g
- sudo apt-get install ksh
- sudo apt-get install unixodbc-dev unixodbc
```

CENTOS/RHEL

```
- sudo yum update
- sudo yum install libaio-devel
- sudo yum install libaio
- sudo yum install glibc
- sudo yum install compat-libstdc++-33
- sudo yum install glibc-devel
- sudo yum install libstdc++
- sudo yum install libstdc++
- sudo yum install pam
- sudo yum install ncurses-devel
- sudo yum install unixODBC
- sudo wget ftp://195.220.108.108/linux/centos/6.7/os/i386/Packages/ksh-20120801-28.el6.i686.rpm
- sudo yum install ksh-20120801-28.el6.i686.rpm
- sudo yum install nano
```

- Download the instant client compatible with the architecture of your operating system, x64 or x86 bits.
 - [DsDriver Linux \(x86 \)](#)
 - [DsDriver Linux \(x64 \)](#)

Installing DsDriver IBM DB2**Architecture x86**

```
sudo mkdir -p /opt/IBM/Db2
sudo cp sudo cp x86_v10.5fp8_linuxia32_dsdriver.tar.gz /opt/IBM/Db2
sudo tar -zxf /opt/IBM/Db2/x86_v10.5fp8_linuxia32_dsdriver.tar.gz
sudo ksh /opt/IBM/Db2/dsdriver/installDSDriver
```

Architecture x64

```
sudo mkdir -p /opt/IBM/x64/db2
sudo cp x64_v10.5fp8_linuxx64_dsdriver.tar.gz /opt/IBM/x64/db2
sudo tar -zxf /opt/IBM/x64/db2/x64_v10.5fp8_linuxx64_dsdriver.tar.gz
sudo ksh /opt/IBM/x64/db2/dsdriver/installDSDriver
sudo source /opt/IBM/x64/db2/dsdriver/db2profile
```

Restart the Apache Server: `sudo /etc/init.d/apachesc9 restart`

Creating a Connection with ScriptCase












- Access ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- After accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- After that, you will see a page with all the database connections that you can create. Select a "DB2" connection

Create a connection
Select your database system and create a connection for your project.

 MySQL	 Oracle	 Microsoft SQL Server	 PostgreSQL	DB2	 Informix
 MS Access	 SQLite	 SyBase	 InterBase	 Firebird	 ODBC

Selecting a Linux database connection

- **new connection** : You will place the information of the connection to your database here:

Create a connection DB2
Enter the connection details for your database DB2

[← Choose another connection](#)

CONNECTION FILTER ADVANCED ?

Connection Name: DBMS Driver:

Server/Host (Name or IP): Port: Database Name: Schema:

Username: Password:

Note: Access the "Filter" tab to define the schema that will be loaded for this connection. Learn more by accessing this [TUTORIAL](#)

Connection Success

[Test Connection](#)

Connecting with Linux database

- **Connection Name** : Defines the name of your new connection.
- **DBMS Driver** : Allows you to select the drivers: DB2 Generic ODBC, DB2 Generic ODBC 6 or lower, DB2, DB2 Native ODBC.
- **Server/Host (Name or IP)** : Insert the name or IP of the server where the Linux database is located.
- **Database Name** : Inform the database name.
- **Schema** : You can inform the Schema (It is optional), in case you want to use the tables that are part of a Schema.
- **Username** : Inform the user to access the Linux database.
- **Password** : Inform the password to access the Linux database.
- **Test Connection** : Displays a message of the status of the connection if successful or not.
- For more options on connecting, click on the **Advanced** tab:

CONNECTION FILTER **ADVANCED** ?

Caution, the settings below are specific to the native ibm_db2 drive. Only change if you know what you are doing. For more details see: http://br2.php.net/manual/pt_BR/function.db2-connect.php

autocommit

i5_lib

i5_naming

i5_commit

i5_query_optimize

Decimal Separator

Persistent Connection

Use the schema before table name

Advanced setup for the Linux database

- **utocommit** - Enables or disables the auto commit functionality.
- **5_lib** - A value of characters that indicates the default library that will be used to solve unqualified file references. This is not valid if the the connection is using the system nomination mode.
- **5_naming** - Enables or disables the system nomination mode of the DB2 UDB CLI iSeries. The files are qualified when using dash (/) delimiter. Unqualified files are solved using the list of libraries to work.
- **5_commit** - Select the desired option treating the date of the IBM DB2.
- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **se the schema before table name** - Allows the use of schema before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER **ADVANCED**

Show

Tables

Views

System tables

Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No ▾

Filtering the Linux database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.



The automatic installation already comes with the Firebird previously configured, only needing to install the client to make the connection.

If you're using your own environment previously configured, you'll need to enable the Firebird extension in PHP.

Firebird

Setting Up FireBird On Windows

The automatic installation already comes with the FireBird driver, only needing to connect with the database in the ScriptCase.

Firebird PDO

Setting Up FireBird PDO On Windows

The automatic installation already comes with the FireBird PDO driver, only needing to connect with the database in the ScriptCase.

Creating a Connection with ScriptCase













- ccess ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- fter accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- fter that, you will see a page with all the database connections that you can create. Select a "FireBird" connection

Create a connection
Select your database system and create a connection for your project.

 MySQL	 Oracle	 MSSQL Server	 Postgres	 DB2	 Informix
 MS Access	 SQLite	 SyBase	 InterBase	 Firebird	 ODBC

Selecting a Windows database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection Firebird

Enter your database connection details for Firebird

[← Choose another connection](#)

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver:

Server/Host (Name or IP): Database Name:

Username: Password:

[Test Connection](#) [Save](#)

Connecting with Windows database

- **Connection Name** : Defines the name of your new connection.
- **DBMS Driver** : Allows you to select the FireBird drivers.
- **Server/Host (Name or IP)** : Insert the name or IP of the server where the Windows database is located.
- **Database Name** : Inform the Database name.
- **Username** : Inform the user to access the Windows database.
- **Password** : Inform the password to access the Windows database.
- **Test Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:

CONNECTION FILTER **ADVANCED**

Decimal Separator:

Persistent Connection:

Use the schema before table name:

[Test Connection](#) [Save](#)

Advanced setup for the Windows database

- **Decimal Separator** - Select the separator type, between a dot or a comma.
- **Persistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **Use the schema before table name** - Allows the use of schema before the name of the tables.
- **Filter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER **ADVANCED**

Show

Tables
 Views
 System tables
 Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	<input type="text" value="No"/>

[Test Connection](#) [Save](#)

Filtering the Windows database

- **Show** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **Tables** - Allows to setup the display of Tables from your database.
 - **Views** - Allows to setup the display of Views from your database.
 - **System Tables** - Allows to setup the display of System Tables from your database.
 - **Procedures** - Allows to setup the display of Procedures from your database.
- **Searches** - Allows to define which tables and Owner will be displayed.
 - **Tables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **Owner** - Allows to define the owner of the tables for listing.
 - **Show** - Allows to define what will be displayed or not from the table an owner's setup.

The automatic installation already comes with the Firebird previously configured, only needing to install the client to make the connection.

If you're using your own environment previously configured, you'll need to enable the Firebird extension in PHP.

Firebird

Setting Up FireBird On Linux

The automatic installation already comes with the FireBird driver, only needing to connect with the database in the ScriptCase.

Firebird PDO

Setting Up FireBird PDO On Linux

The automatic installation already comes with the FireBird PDO driver, only needing to connect with the database in the ScriptCase.

Creating a Connection with ScriptCase

- ccess ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- fter accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"















Creating a new connection

- fter that, you will see a page with all the database connections that you can create. Select a "FireBird" connection

Create a connection

Select your database system and create a connection for your project.

 MySQL	 Oracle	 MSSQL Server	 Postgres	 DB2	 Informix
 MS Access	 SQLite	 SyBase	 InterBase	 Firebird	 ODBC

Selecting a Linux database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection Firebird

Enter your database connection details for Firebird

[← Choose another connection](#)

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver:

Server/Host (Name or IP): Database Name:

Username: Password:

[Test Connection](#) [Save](#)

Connecting with Linux database

- **Connection Name** : Defines the name of your new connection.
- **DBMS Driver** : Allows you to select the FireBird drivers.
- **Server/Host (Name or IP)** : Insert the name or IP of the server where the Linux database is located.
- **Database Name** : Inform the Database name.
- **Username** : Inform the user to access the Linux database.
- **Password** : Inform the password to access the Linux database.
- **Test Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:

CONNECTION FILTER **ADVANCED**

Decimal Separator:

Persistent Connection:

Use the schema before table name:

[Test Connection](#) [Save](#)

Advanced setup for the Linux database

- **Decimal Separator** - Select the separator type, between a dot or a comma.
- **Persistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **Use the schema before table name** - Allows the use of schema before the name of the tables.
- **Filter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER **ADVANCED**

Show

Tables
 Views
 System tables
 Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	<input type="text" value="No"/>

[Test Connection](#) [Save](#)

Filtering the Linux database

- **Show** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **Tables** - Allows to setup the display of Tables from your database.
 - **Views** - Allows to setup the display of Views from your database.
 - **System Tables** - Allows to setup the display of System Tables from your database.
 - **Procedures** - Allows to setup the display of Procedures from your database.
- **Searches** - Allows to define which tables and Owner will be displayed.
 - **Tables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **Owner** - Allows to define the owner of the tables for listing.
 - **Show** - Allows to define what will be displayed or not from the table an owner's setup.

The automatic installation already comes with the Firebird previously configured, only needing to install the client to make the connection.

If you're using your own environment previously configured, you'll need to enable the Firebird extension in PHP.

Firebird

Setting Up FireBird On MAC

The automatic installation already comes with the FireBird previously configured, only needing to install the client so that you can do the connection.

Previous Requirements

- download the installation: [Client FireBird](#)
- install the client.
- restart the Apache Server.

Firebird PDO

Setting Up FireBird PDO On MAC

The automatic installation already comes with the FireBird PDO driver, only needing to connect with the database in the ScriptCase.

Pre-Requirements

- download the installation: [Client InterBase](#)

Restart the Apache Server:

- `applications/NetMake/v9/components/xamppfiles/xampp restart`

Creating a Connection with ScriptCase

- access ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- after accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"















Creating a new connection

- after that, you will see a page with all the database connections that you can create. Select a "FireBird" connection

Create a connection

Select your database system and create a connection for your project.

 MySQL	 Oracle	 MSSQL Server	 Postgres	 DB2	 Informix
 MS Access	 SQLite	 SyBase	 InterBase	 Firebird	 ODBC

Selecting a Mac OS X database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection Firebird

Enter your database connection details for Firebird

← Choose another connection

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver:

Server/Host (Name or IP): Database Name:

Username: Password:

Connecting with Mac OS X database

- **onnection Name** : Defines the name of your new connection.
- **BMS Driver** : Allows you to select the FireBird drivers.
- **erver/Host (Name or IP)** : Insert the name or IP of the server where the Mac OS X database is located.
- **atabase Name** : Inform the Database name.
- **ername** : Inform the user to access the Mac OS X database.
- **assword** : Inform the password to access the Mac OS X database.
- **est Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:

CONNECTION FILTER **ADVANCED**

Decimal Separator:

Persistent Connection:

Use the schema before table name:

Advanced setup for the Mac OS X database

- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **se the schema before table name** - Allows the use of schema before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER ADVANCED

Show

Tables

Views

System tables

Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No ▾

Test Connection
Save

Filtering the Mac OS X database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.



Currently ScriptCase offers the connection to the Informix through the SDK. The Informix extension comes disabled in the installer.

If you are using your own previously configured environment, you will need to enable the Informix extension in PHP.

Informix PDO

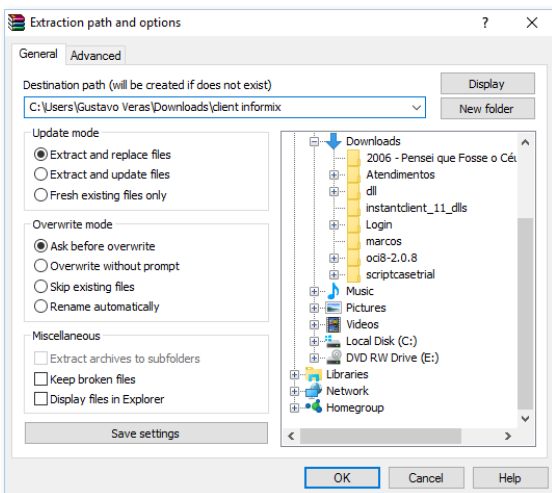
Setting Up Informix On Windows

The automatic installation already comes with the PDO Informix previously configured, only needing to install the client so that you can do the connection.

- download the instant client compatible with the architecture of your operating system, x64 or x86 bits:
 - [ClienteSDK Windows \(x86 \)](#)
 - [ClienteSDK Windows \(x64 \)](#)

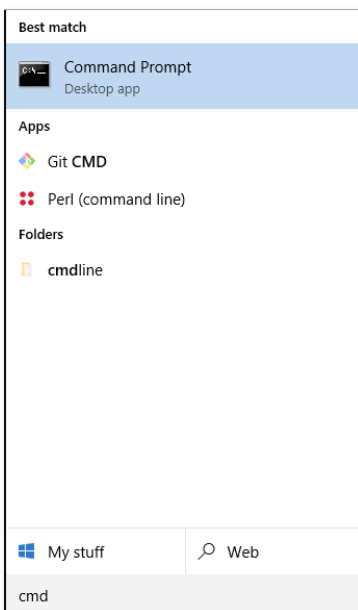
Installing ClienteSDK Informix

- unzip the file into any folder. e.g.: **client informix**



Unzipping the file.

- access the **CMD** from your windows by searching for a command prompt in Windows Search or by pricing **ctrl + r** and typing **cmd** in the search.



Accessing Windows Command Prompt.

- when starting the **CMD**, access the folder where the file was saved after the download using the **cd** command and informing the path. Example:

```
C:\WINDOWS\system32\cmd.exe
C:\Users\alvar>cd "Downloads\client_informix"
```

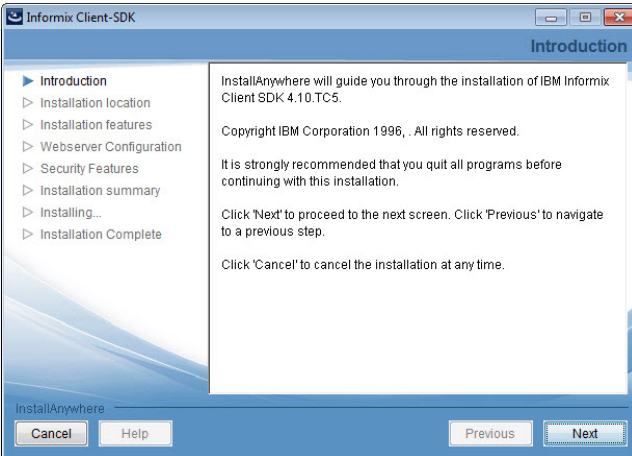
Accessing the client folder.

- Then run the .exe file with the -i and GUI parameters. Example: installclientsdk.exe -i GUI_

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [versão 10.0.18586]
(c) 2015 Microsoft Corporation. Todos os direitos reservados.
C:\Users\alvar>cd "Downloads\client_informix"
C:\Users\alvar>Downloads\client_informix>installclientsdk.exe -i GUI
```

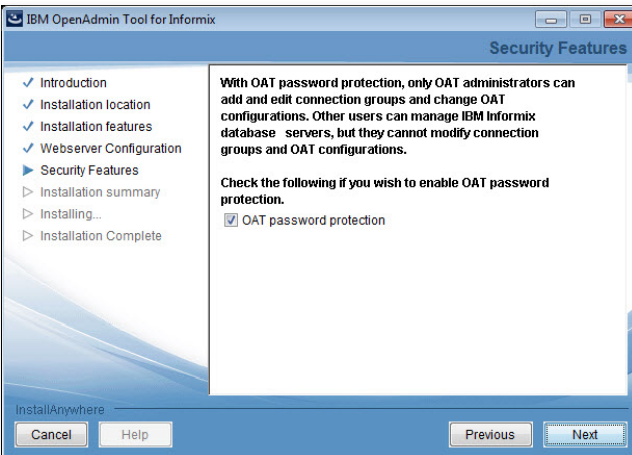
Running the installer.

- Follow the client installation wizard, clicking next, to maintain the default installation values. Pay attention to the Security Features step.



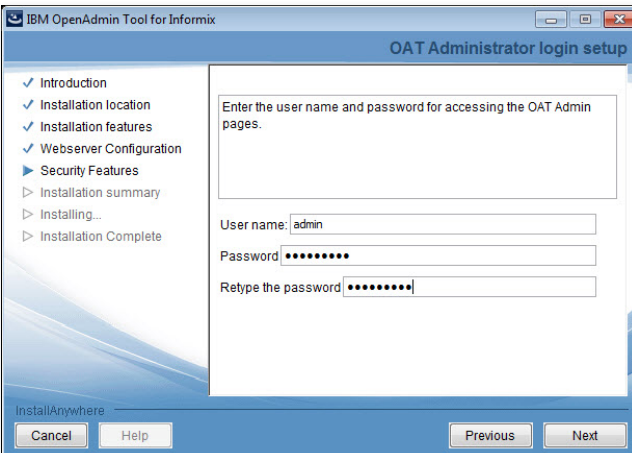
Installation Wizard.

- At this point you can choose whether or not to use a password to access the Informix Administration Application (OTA)



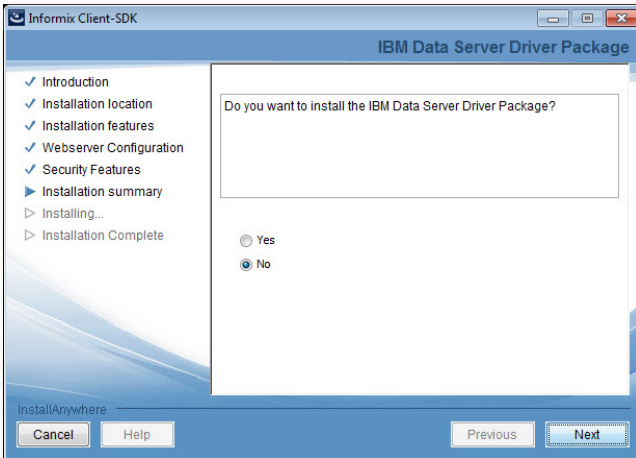
OTA Definition.

- Set your login and password, if you have checked the **OAT password protection**



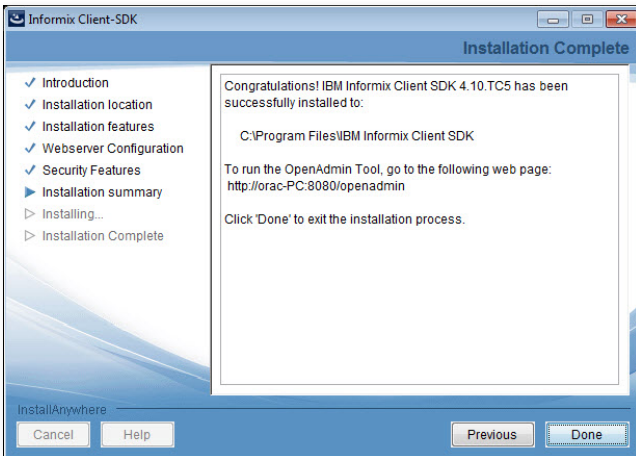
OTA password Definition.

- elect **NO** and proceed with the installation normally.



Installation Wizard.

- proceed with the installation by clicking **Done** and wait for the process to end.



Finish the Installation.

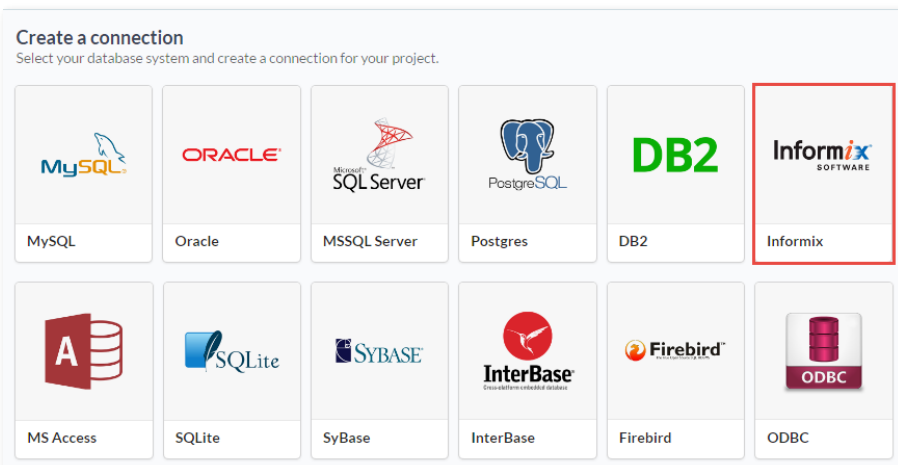
Creating a Connection with ScriptCase

- access ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- after accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- after that, you will see a page with all the database connections that you can create. Select a "Informix" connection



Selecting a Windows database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection Informix
Enter your database connection details for Informix

[← Choose another connection](#)

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver:

Server/Host (Name or IP): Database Name:

Username: Password:

[Test Connection](#) [Save](#)

Connecting with Windows database

- **Connection Name** : Defines the name of your new connection.
- **DBMS Driver** : Allows you to select the Informix PDO drivers.
- **Server/Host (Name or IP)** : Insert the name or IP of the server where the Windows database is located.
- **Database Name** : Inform name of the database.
- **Username** : Inform the user to access the Windows database.
- **Password** : Inform the password to access the Windows database.
- **Test Connection** : Displays a message of the status of the connection if successful or not.
- For more options on connecting, click on the **Advanced** tab:

CONNECTION FILTER **ADVANCED**

Decimal Separator:

Persistent Connection:

Use the schema before table name:

[Test Connection](#) [Save](#)

Advanced setup for the Windows database

- **Decimal Separator** - Select the separator type, between a dot or a comma.
- **Persistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **Use the schema before table name** - Allows the use of schema before the name of the tables.
- **Filter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER **ADVANCED**

Show

Tables
 Views
 System tables
 Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	<input type="text" value="No"/>

[Test Connection](#) [Save](#)

Filtering the Windows database

- **Show** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **Tables** - Allows to setup the display of Tables from your database.
 - **Views** - Allows to setup the display of Views from your database.
 - **System Tables** - Allows to setup the display of System Tables from your database.
 - **Procedures** - Allows to setup the display of Procedures from your database.
- **Searches** - Allows to define which tables and Owner will be displayed.
 - **Tables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **Owner** - Allows to define the owner of the tables for listing.
 - **Show** - Allows to define what will be displayed or not from the table an owner's setup.

Currently ScriptCase offers the connection to the Informix through the SDK. The Informix extension comes disabled in the installer.

If you are using your own previously configured environment, you will need to enable the Informix extension in PHP.

Informix PDO

Setting Up Informix On Linux

The automatic installation already comes with the PDO Informix previously configured, only needing to install the client so that you can do the connection.

- install the dependencies below:

UBUNTUDEBIAN

```
- sudo apt-get update
- sudo apt-get install libaio1
- sudo apt-get install libncurses5
- sudo apt-get install alien
- sudo apt-get install gcc-multilib g++-multilib
- sudo apt-get install libpam0g
- sudo apt-get install ksh
- sudo apt-get install unixodbc-dev unixodbc
```

CENTOSRHEL

```
- sudo yum update
- sudo yum install libaio-devel
- sudo yum install libaio
- sudo yum install glibc
- sudo yum install compat-libstdc++-33
- sudo yum install glibc-devel
- sudo yum install libstdc++
- sudo yum install libstdc++
- sudo yum install pam
- sudo yum install ncurses-devel
- sudo yum install unixODBC
- sudo wget ftp://195.220.108.108/linux/centos/6.7/os/i386/Packages/ksh-20120801-28.el6.i686.rpm
- sudo yum install ksh-20120801-28.el6.i686.rpm
- sudo yum install nano
```

- download the instant client compatible with the architecture of your operating system, x64 or x86 bits:
 - [ClienteSDK Linux \(x86 \)](#)
 - [ClienteSDK Linux \(x64 \)](#)

Installing ClienteSDK Informix

Architecture x86

```
sudo tar -xf x86_clientsdk.3.70.UC8DE.LINUX.tar
```

Note: The Informix client installation directory should be: /opt/IBM/ixf

```
sudo ./installclientsdk
```

Selecione 1 para aceitar os termos.

Selecione as opções: 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17

Prossiga com a instalação.

Architecture x64

```
sudo tar -xf x64_clientsdk.4.10.FC6DE.LINUX.tar
```

Note: The Informix client installation directory should be: /opt/IBM/x64/informix

```
sudo ./installclientsdk
```

Selecione 1 para aceitar os termos.

Selecione as opções: 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17

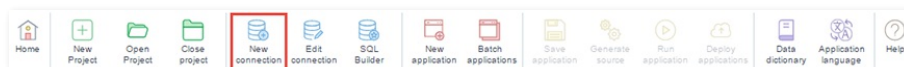
Prossiga com a instalação.

Restart the Apache Server:

```
sudo /etc/init.d/apachesc9 restart
```

Creating a Connection with ScriptCase

- access ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- after accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- after that, you will see a page with all the database connections that you can create. Select a "Informix" connection

Create a connection
Select your database system and create a connection for your project.

MySQL Oracle MSSQL Server Postgres DB2 Informix
MS Access SQLite SyBase InterBase Firebird ODBC

Selecting a Linux database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection Informix
Enter your database connection details for Informix [← Choose another connection](#)

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver:

Server/Host (Name or IP): Database Name:

Username: Password:

[Test Connection](#) [Save](#)

Connecting with Linux database

- **onnection Name** : Defines the name of your new connection.
- **BMS Driver** : Allows you to select the Informix PDO drivers.
- **erver/Host (Name or IP)** : Insert the name or IP of the server where the Linux database is located.
- **atabase Name** : Inform name of the database.
- **ername** : Inform the user to access the Linux database.
- **assword** : Inform the password to access the Linux database.
- **est Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:

CONNECTION FILTER **ADVANCED**

Decimal Separator:

Persistent Connection:

Use the schema before table name:

[Test Connection](#) [Save](#)

Advanced setup for the Linux database

- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **se the schema before table name** - Allows the use of schema before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER ADVANCED

Show

Tables

Views

System tables

Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No ▾

Test Connection
Save

Filtering the Linux database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.

Currently ScriptCase offers the connection to the Informix through the SDK. The Informix extension comes disabled in the installer.

If you are using your own previously configured environment, you will need to enable the Informix extension in PHP.

Informix PDO

Setting Up Informix On MAC

The automatic installation already comes with the PDO Informix previously configured, only needing to install the client so that you can do the connection.

- download the instant client
 - [ClientSDK Mac OS X](#)
- extract the .tar file
- o the installation through the terminal, accessing the extracted folder and executing the command below:
 - `udo ./installclientsdk`
- **install in the default directory (/Applications/IBM/informix)**
- choose options from 1 to 17
- restart the web server.

Creating a Connection with ScriptCase













- access ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- after accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- after that, you will see a page with all the database connections that you can create. Select a "Informix" connection

Create a connection
Select your database system and create a connection for your project.

 MySQL	 Oracle	 Microsoft SQL Server	 Postgres	 DB2	 Informix
 MS Access	 SQLite	 SyBase	 InterBase	 Firebird	 ODBC

Selecting a Mac OS X database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection Informix

Enter your database connection details for Informix

[← Choose another connection](#)

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver:

Server/Host (Name or IP): Database Name:

Username: Password:

[Test Connection](#) [Save](#)

Connecting with Mac OS X database

- **Connection Name** : Defines the name of your new connection.
- **DBMS Driver** : Allows you to select the Informix PDO drivers.
- **Server/Host (Name or IP)** : Insert the name or IP of the server where the Mac OS X database is located.
- **Database Name** : Inform name of the database.
- **Username** : Inform the user to access the Mac OS X database.
- **Password** : Inform the password to access the Mac OS X database.
- **Test Connection** : Displays a message of the status of the connection if successful or not.
- For more options on connecting, click on the **Advanced** tab:

CONNECTION FILTER **ADVANCED**

Decimal Separator:

Persistent Connection:

Use the schema before table name:

[Test Connection](#) [Save](#)

Advanced setup for the Mac OS X database

- **Decimal Separator** - Select the separator type, between a dot or a comma.
- **Persistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **Use the schema before table name** - Allows the use of schema before the name of the tables.
- **Filter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER **ADVANCED**

Show

Tables
 Views
 System tables
 Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	<input type="text" value="No"/>

[Test Connection](#) [Save](#)

Filtering the Mac OS X database

- **Show** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **Tables** - Allows to setup the display of Tables from your database.
 - **Views** - Allows to setup the display of Views from your database.
 - **System Tables** - Allows to setup the display of System Tables from your database.
 - **Procedures** - Allows to setup the display of Procedures from your database.
- **Searches** - Allows to define which tables and Owner will be displayed.
 - **Tables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **Owner** - Allows to define the owner of the tables for listing.
 - **Show** - Allows to define what will be displayed or not from the table an owner's setup.



The automatic installation already comes with the InterBase extension only needing to connect with the database.

If you are using your own previously configured environment, you will need to enable the InterBase extension in PHP.

InterBase 6.5 or Higher

Setting Up InterBase 6 Or Higher On Windows

The automatic installation already comes with the FireBird driver, only needing to connect with the database in the ScriptCase.

InterBase

Setting Up InterBase On Windows

The automatic installation already comes with the FireBird driver, only needing to connect with the database in the ScriptCase.

Creating a Connection with ScriptCase













- ccess ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- fter accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- fter that, you will see a page with all the database connections that you can create. Select a "InterBase" connection

Create a connection
Select your database system and create a connection for your project.

 MySQL	 Oracle	 MSSQL Server	 Postgres	 DB2	 Informix
 MS Access	 SQLite	 SyBase	 InterBase	 Firebird	 ODBC

Selecting a Windows database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection InterBase

Enter your database connection details for InterBase

[← Choose another connection](#)

CONNECTION FILTER ADVANCED

Connection Name DBMS Driver

IP:PATH

Username Password

[Test Connection](#) [Save](#)

Connecting with Windows database

- **Connection Name** : Defines the name of your new connection.
- **DBMS Driver** : Allows you to select the Interbase 6.5 or higher drivers.
- **P:PATH** : Inform the IP and path of the server to the location of the Windows database.
 - **PATH** : Path of the Windows database.
- **username** : Inform the user to access the Windows database.
- **password** : Inform the password to access the Windows database.
- **Test Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:

CONNECTION FILTER **ADVANCED**

Decimal Separator

Persistent Connection

Use the schema before table name

[Test Connection](#) [Save](#)

Advanced setup for the Windows database

- **Decimal Separator** - Select the separator type, between a dot or a comma.
- **Persistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **Use the schema before table name** - Allows the use of schema before the name of the tables.
- **Filter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER **ADVANCED**

Show

Tables

Views

System tables

Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	<input type="text" value="No"/>

[Test Connection](#) [Save](#)

Filtering the Windows database

- **Show** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **Tables** - Allows to setup the display of Tables from your database.
 - **Views** - Allows to setup the display of Views from your database.
 - **System Tables** - Allows to setup the display of System Tables from your database.
 - **Procedures** - Allows to setup the display of Procedures from your database.
- **Searches** - Allows to define which tables and Owner will be displayed.
 - **Tables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **Owner** - Allows to define the owner of the tables for listing.
 - **Show** - Allows to define what will be displayed or not from the table an owner's setup.

The automatic installation already comes with the InterBase extension only needing to connect with the database.

If you are using your own previously configured environment, you will need to enable the InterBase extension in PHP.

InterBase 6.5 or Higher

Setting Up InterBase 6 Or Higher On Linux

The automatic installation already comes with the FireBird driver, only needing to connect with the database in the ScriptCase.

InterBase

Setting Up InterBase On Linux

The automatic installation already comes with the FireBird driver, only needing to connect with the database in the ScriptCase.

Creating a Connection with ScriptCase













- ccess ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- fter accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- fter that, you will see a page with all the database connections that you can create. Select a "InterBase" connection

Create a connection
Select your database system and create a connection for your project.

 MySQL	 Oracle	 MSSQL Server	 Postgres	 DB2	 Informix
 MS Access	 SQLite	 SyBase	 InterBase	 Firebird	 ODBC

Selecting a Linux database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection InterBase
Enter your database connection details for InterBase

[← Choose another connection](#)

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver:

IP:PATH:

Username: Password:

[Test Connection](#) [Save](#)

Connecting with Linux database

- **Connection Name** : Defines the name of your new connection.
- **DBMS Driver** : Allows you to select the Interbase 6.5 or higher drivers.
- **P:PATH** : Inform the IP and path of the server to the location of the Linux database.
 - **PATH** : Path of the Linux database.
- **username** : Inform the user to access the Linux database.
- **password** : Inform the password to access the Linux database.
- **Test Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:

CONNECTION FILTER **ADVANCED**

Decimal Separator:

Persistent Connection:

Use the schema before table name:

[Test Connection](#) [Save](#)

Advanced setup for the Linux database

- **Decimal Separator** - Select the separator type, between a dot or a comma.
- **Persistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **Use the schema before table name** - Allows the use of schema before the name of the tables.
- **Filter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER **ADVANCED**

Show

Tables
 Views
 System tables
 Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	<input type="text" value="No"/>

[Test Connection](#) [Save](#)

Filtering the Linux database

- **Show** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **Tables** - Allows to setup the display of Tables from your database.
 - **Views** - Allows to setup the display of Views from your database.
 - **System Tables** - Allows to setup the display of System Tables from your database.
 - **Procedures** - Allows to setup the display of Procedures from your database.
- **Searches** - Allows to define which tables and Owner will be displayed.
 - **Tables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **Owner** - Allows to define the owner of the tables for listing.
 - **Show** - Allows to define what will be displayed or not from the table an owner's setup.

The automatic installation already comes with the InterBase extension only needing to connect with the database.

If you are using your own previously configured environment, you will need to enable the InterBase extension in PHP.

InterBase 6.5 or Higher

Setting Up InterBase 6 Or Higher On MAC

The automatic installation already comes with the Interbase previously configured, only needing to install the client so that you can do the connection.

Previous Requirements

- download the installation: [Client InterBase](#)

Restart the Apache Server:

- `applications/NetMake/v9/components/xamppfiles/xampp restart`

InterBase

Setting Up InterBase On MAC

The automatic installation already comes with the FireBird driver, only needing to connect with the database in the ScriptCase.

Pre-Requirements

- download the installation: [Client InterBase](#)

Restart the Apache Server:

- `applications/NetMake/v9/components/xamppfiles/xampp restart`

Creating a Connection with ScriptCase

- access ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- after accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"

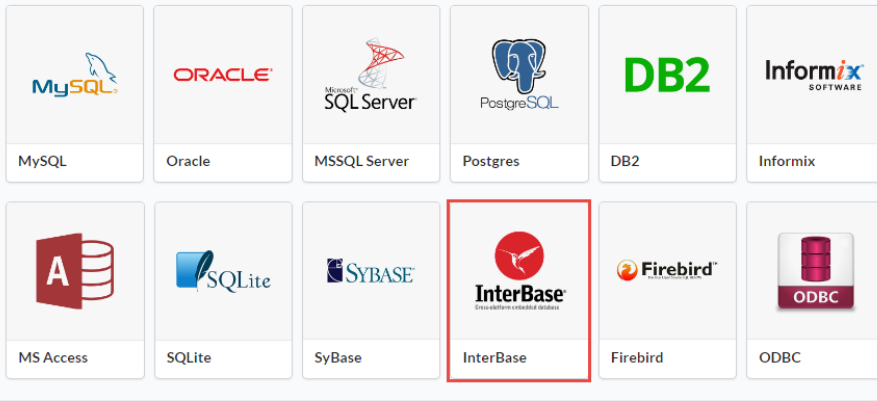


Creating a new connection

- after that, you will see a page with all the database connections that you can create. Select a "InterBase" connection

Create a connection

Select your database system and create a connection for your project.



Selecting a Mac OS X database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection InterBase

Enter your database connection details for InterBase ← Choose another connection

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver:

IP:PATH:

Username: Password:

Connecting with Mac OS X database

- **onnection Name** : Defines the name of your new connection.
- **BMS Driver** : Allows you to select the Interbase 6.5 or higher drivers.
- **P:PATH** : Inform the IP and path of the server to the location of the Mac OS X database.
 - **ATH** : Path of the Mac OS X database.
- **ername** : Inform the user to access the Mac OS X database.
- **assword** : Inform the password to access the Mac OS X database.
- **est Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:

CONNECTION FILTER **ADVANCED**

Decimal Separator:

Persistent Connection:

Use the schema before table name:

Advanced setup for the Mac OS X database

- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **se the schema before table name** - Allows the use of schema before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER ADVANCED

Show

Tables

Views

System tables

Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No ▾

Test Connection
Save

Filtering the Mac OS X database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.



The automatic installation already comes with the MySQL extension only needing to connect with the database.

If you are using your own previously configured environment, you will need to enable the MySQL extension in PHP.

Server sent charset unknown to the client

If you are using MySQL 8, due to the PHP 7 bug, an error occurs when trying to connect to the ScriptCase.

[Click here](#) and see how to work around this error.

MySQL PDO

Setting Up MySQL PDO On Windows

The automatic installation already comes with the MySQL driver, only needing to connect with the database with ScriptCase.

MySQLi

Setting Up MySQLi On Windows

The automatic installation already comes with the MySQL driver, only needing to connect with the database with ScriptCase.

Creating a Connection with ScriptCase

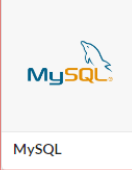
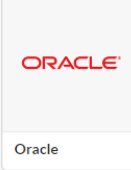
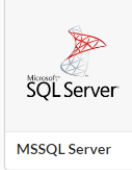


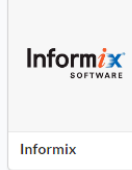
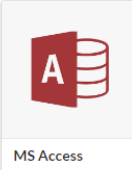

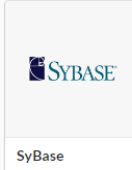
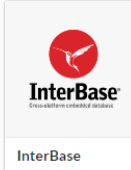
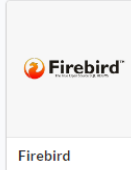
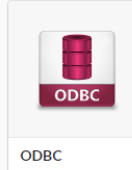
- ccess ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- fter accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- fter that, you will see a page with all the database connections that you can create. Select a "MySQL" connection

Create a connection
Select your database system and create a connection for your project.

 MySQL	 Oracle	 MSSQL Server	 Postgres	 DB2	 Informix
 MS Access	 SQLite	 SyBase	 InterBase	 Firebird	 ODBC

Selecting a Windows database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection MySQL

Enter your database connection details for MySQL

← Choose another connection

CONNECTION SECURITY FILTER ADVANCED

Connection Name: DBMS Driver:

Server/Host (Name or IP): Port:

Username: Password:

Database Name:

Connecting with Windows database

- **Connection Name** : Defines the name of your new connection.
- **DBMS Driver** : Allows you to select the drivers: MySQL PDO.
- **Server/Host (Name or IP)** : Insert the name or IP of the server where the Windows database is located.
- **Port** : Define the port used for the connection. By default it is 3306.
- **Database Name** : Select the database.
- **Username** : Inform the user to access the Windows database.
- **Password** : Inform the password to access the Windows database.
- **Test Connection** : Displays a message of the status of the connection if successful or not.

Create a connection MySQL

Enter your database connection details for MySQL

← Choose another connection

CONNECTION SECURITY FILTER ADVANCED

Use SSL

Client Key:

Client Certificate:

CA Path:

CA Certificate:

Specified Cipher:

Setting up security with Windows

- **Use SSL** : Activates or deactivates the use of secure connections with Windows.
- **Client Key** : Path to the client's private identification key in the format PEM.
- **Client Certificate** : Path to the client's public certificate key.
- **CA Path** : Path to the directory that contains the Authority Certification (AC) in the format PEM, if used, it should specify the same certificate used by the server.
- **CA Certificate** : Path to the Authority Certificate (AC) in the format PEM. This option, if used, should specify the same certificate as the server.
- **Specified Cipher** : A list of permitted numbers to use with the cryptography of the connection. If one of the numbers is not supported, the SSL connection won't work.

For more details, consult: [Setting up MySQL to use secure connections](#)

- For more options on connecting, click on the **Advanced** tab:

Create a connection MySQL

Enter your database connection details for MySQL

← Choose another connection

CONNECTION SECURITY FILTER **ADVANCED**

client_encoding
utf8

Decimal Separator
.

Persistent Connection
No

Use the schema before table name
Yes

Test Connection Save

Advanced setup for the Windows database

- **lient_encoding** : Database Encoding.
- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **se the schema before table name** - Allows the use of schemas before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION **FILTER** ADVANCED

Show

Tables
 Views
 System tables
 Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No

Test Connection Save

Filtering the Windows database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.

The automatic installation already comes with the MySQL extension only needing to connect with the database.

If you are using your own previously configured environment, you will need to enable the MySQL extension in PHP.

Server sent charset unknown to the client

If you are using MySQL 8, due to the PHP 7 bug, an error occurs when trying to connect to the ScriptCase.

[Click here](#) and see how to work around this error.

MySQL PDO

Setting Up MySQL PDO On Linux

The automatic installation already comes with the MySQL driver, only needing to connect with the database with ScriptCase.

MySQLi

Setting Up MySQLi On Linux

The automatic installation already comes with the MySQL driver, only needing to connect with the database with ScriptCase.

Creating a Connection with ScriptCase

- ccess ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- fter accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"

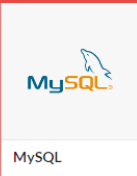
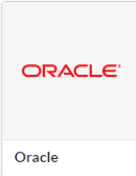
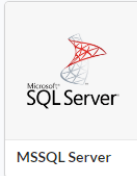

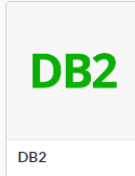
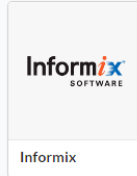
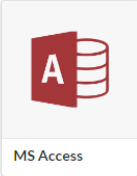

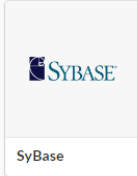
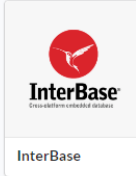
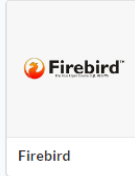
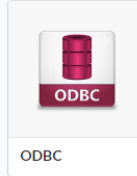


Creating a new connection

- fter that, you will see a page with all the database connections that you can create. Select a "MySQL" connection

Create a connection

Select your database system and create a connection for your project.

 MySQL	 Oracle	 MSSQL Server	 Postgres	 DB2	 Informix
 MS Access	 SQLite	 SyBase	 InterBase	 Firebird	 ODBC

Selecting a Linux database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection MySQL

Enter your database connection details for MySQL

← Choose another connection

CONNECTION SECURITY FILTER ADVANCED

Connection Name: DBMS Driver:

Server/Host (Name or IP): Port:

Username: Password:

Database Name:

Connecting with Linux database

- **Connection Name** : Defines the name of your new connection.
- **DBMS Driver** : Allows you to select the drivers: MySQL PDO.
- **Server/Host (Name or IP)** : Insert the name or IP of the server where the Linux database is located.
- **Port** : Define the port used for the connection. By default it is 3306.
- **Database Name** : Select the database.
- **Username** : Inform the user to access the Linux database.
- **Password** : Inform the password to access the Linux database.
- **Test Connection** : Displays a message of the status of the connection if successful or not.

Create a connection MySQL

Enter your database connection details for MySQL

← Choose another connection

CONNECTION SECURITY FILTER ADVANCED

Use SSL

Client Key:

Client Certificate:

CA Path:

CA Certificate:

Specified Cipher:

Setting up security with Linux

- **Use SSL** : Activates or deactivates the use of secure connections with Linux.
- **Client Key** : Path to the client's private identification key in the format PEM.
- **Client Certificate** : Path to the client's public certificate key.
- **CA Path** : Path to the directory that contains the Authority Certification (AC) in the format PEM, if used, it should specify the same certificate used by the server.
- **CA Certificate** : Path to the Authority Certificate (AC) in the format PEM. This option, if used, should specify the same certificate as the server.
- **Specified Cipher** : A list of permitted numbers to use with the cryptography of the connection. If one of the numbers is not supported, the SSL connection won't work.

For more details, consult: [Setting up MySQL to use secure connections](#)

- or more options on connecting, click on the **Advanced** tab:

Create a connection MySQL

Enter your database connection details for MySQL

← Choose another connection

CONNECTION SECURITY FILTER **ADVANCED**

client_encoding
utf8

Decimal Separator
.

Persistent Connection
No

Use the schema before table name
Yes

Test Connection Save

Advanced setup for the Linux database

- **lient_encoding** : Database Encoding.
- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **se the schema before table name** - Allows the use of schemas before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION **FILTER** ADVANCED

Show

Tables
 Views
 System tables
 Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No

Test Connection Save

Filtering the Linux database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.

The automatic installation already comes with the MySQL extension only needing to connect with the database.

If you are using your own previously configured environment, you will need to enable the MySQL extension in PHP.

Server sent charset unknown to the client

If you are using MySQL 8, due to the PHP 7 bug, an error occurs when trying to connect to the ScriptCase.

[Click here](#) and see how to work around this error.

MySQL PDO

Setting Up MySQL PDO On MAC

The automatic installation already comes with the MySQL driver, only needing to connect with the database with ScriptCase.

MySQLi

Setting Up MySQLi On MAC

The automatic installation already comes with the MySQL driver, only needing to connect with the database with ScriptCase.

Creating a Connection with ScriptCase

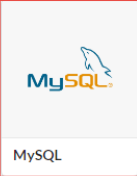
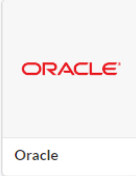
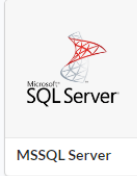

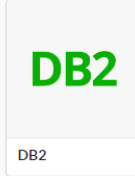
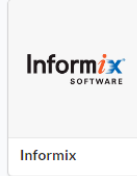
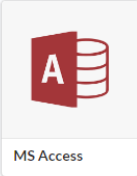

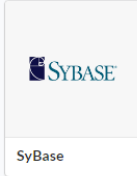
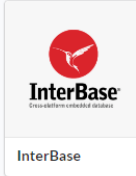
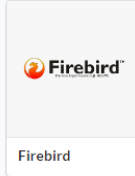
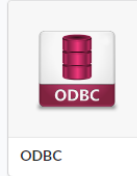
- ccess ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- fter accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- fter that, you will see a page with all the database connections that you can create. Select a "MySQL" connection

Create a connection
Select your database system and create a connection for your project.

 MySQL	 Oracle	 MSSQL Server	 Postgres	 DB2	 Informix
 MS Access	 SQLite	 SyBase	 InterBase	 Firebird	 ODBC

Selecting a Mac OS X database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection MySQL
Enter your database connection details for MySQL

← Choose another connection

CONNECTION SECURITY FILTER ADVANCED

Connection Name: DBMS Driver:

Server/Host (Name or IP): Port:

Username: Password:

Database Name:

Connecting with Mac OS X database

- **Connection Name** : Defines the name of your new connection.
- **DBMS Driver** : Allows you to select the drivers: MySQL PDO.
- **Server/Host (Name or IP)** : Insert the name or IP of the server where the Mac OS X database is located.
- **Port** : Define the port used for the connection. By default it is 3306.
- **Database Name** : Select the database.
- **Username** : Inform the user to access the Mac OS X database.
- **Password** : Inform the password to access the Mac OS X database.
- **Test Connection** : Displays a message of the status of the connection if successful or not.

Create a connection MySQL
Enter your database connection details for MySQL

← Choose another connection

CONNECTION SECURITY FILTER ADVANCED

Use SSL

Client Key:

Client Certificate:

CA Path:

CA Certificate:

Specified Cipher:

Setting up security with Mac OS X

- **Use SSL** : Activates or deactivates the use of secure connections with Mac OS X.
- **Client Key** : Path to the client's private identification key in the format PEM.
- **Client Certificate** : Path to the client's public certificate key.
- **CA Path** : Path to the directory that contains the Authority Certification (AC) in the format PEM, if used, it should specify the same certificate used by the server.
- **CA Certificate** : Path to the Authority Certificate (AC) in the format PEM. This option, if used, should specify the same certificate as the server.
- **Specified Cipher** : A list of permitted numbers to use with the cryptography of the connection. If one of the numbers is not supported, the SSL connection won't work.

For more details, consult: [Setting up MySQL to use secure connections](#)

- or more options on connecting, click on the **Advanced** tab:

Create a connection MySQL

Enter your database connection details for MySQL

← Choose another connection

CONNECTION SECURITY FILTER **ADVANCED**

client_encoding
utf8

Decimal Separator
.

Persistent Connection
No

Use the schema before table name
Yes

Test Connection Save

Advanced setup for the Mac OS X database

- **lient_encoding** : Database Encoding.
- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **se the schema before table name** - Allows the use of schemas before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION **FILTER** ADVANCED

Show

Tables
 Views
 System tables
 Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No

Test Connection Save

Filtering the Mac OS X database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.



The automatic installation already comes with the Oracle extension only needing to connect with the database. If you are using your own previously configured environment, you will need to enable the Oracle extension in PHP.

To enable this extension, we ask that you install the Visual C ++ 2010 x86 and x64 libraries.

Oracle PDO

Setting Up Oracle PDO On Windows

The automatic installation already comes with the Oracle PDO preconfigured, only needing to install the client so that you can do the connection.

*Enter Oracle's website (www.oracle.com) and download **Instant Client Basic** according to the architecture of your operating system (x64 or x86)

Note: PHP 7.0 is only compatible with instant client 11 or higher

Version 12.1.0.2.0

Instant Client Package - Basic: All files required to run OCI, OCCI, and JDBC-OCI applications
[instantclient-basic-nt-12.1.0.2.0.zip](#) (67,410,370 bytes) (cksum - 530881682)

*Instant Client Package - Basic Lite: Smaller version of the Basic, with only English error messages and Unicode, ASCII, and Western European character set support
[instantclient-basictlite-nt-12.1.0.2.0.zip](#) (33,851,306 bytes) (cksum - 3971707995)

*Instant Client Package - JDBC Supplement: Additional support for XA, Internationalization, and RowSet operations under JDBC
[instantclient-jdbc-nt-12.1.0.2.0.zip](#) (1,573,642 bytes) (cksum - 2509633653)

*Instant Client Package - SQL*Plus: Additional libraries and executable for running SQL*Plus with Instant Client
[instantclient-sqlplus-nt-12.1.0.2.0.zip](#) (805,513 bytes) (cksum - 2478532846)

*Instant Client Package - SDK: Additional header files and an example makefile for developing Oracle applications with Instant Client
[instantclient-sdk-nt-12.1.0.2.0.zip](#) (1,951,770 bytes) (cksum - 722480528)

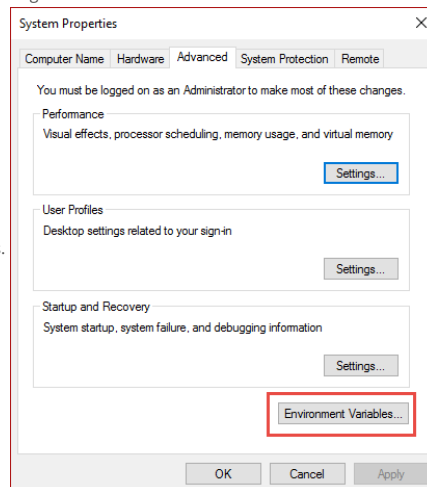
*Instant Client Package - ODBC: Additional libraries for enabling ODBC applications
[instantclient-odbc-nt-12.1.0.2.0.zip](#) (505,668 bytes) (cksum - 41700632)

*Instant Client Package - WRC: Workload Replay Client used to replay workload for RAT's DB Replay Feature
[instantclient-tools-nt-12.1.0.2.0.zip](#) (15,225 bytes) (cksum - 2411353204)

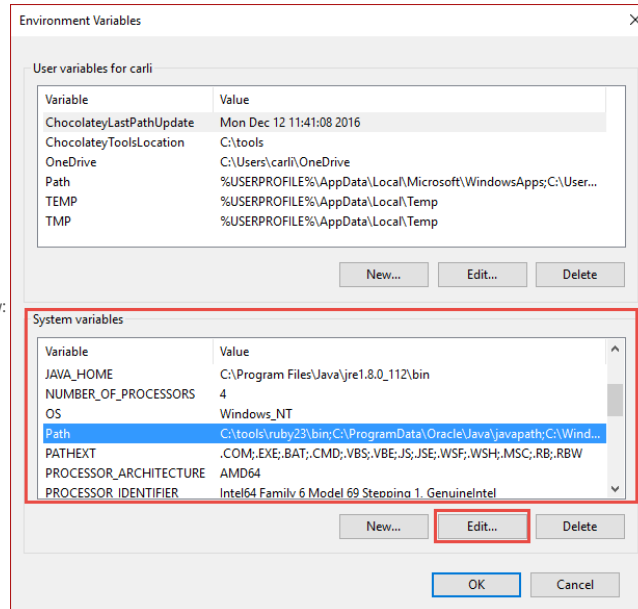
Download Instant Client 12.

- xtract the **instant client basic** package from your computer (e.g.: C:\instantclient_12_1).
- e should put in the folder **PHP(NetMake/v9/components/php/)** the following DLLs:
 - opy the dll **oraociei12.dll** (it is inside the folder of the instant client (e.g.: C:\instantclient_12_1)).
- dd to the system PATH the path of this folder:
 - o to Control Panel> System> Advanced System Settings.

- n the Advanced tab> Click Environment Variables.



Advanced Settings.



- o dd the path as shown below:

Adding client path.

- restart the web server

Oracle 8.0.5 or Higher

Setting Up Oracle 8.0.5 Or Higher On Windows

The automatic installation already comes with the Oracle preconfigured, only needing to install the client so that you can do the connection.

*Enter Oracle's website (www.oracle.com) and download **Instant Client Basic** according to the architecture of your operating system (x64 or x86)

Note: PHP 7.0 is only compatible with instant client 11 or higher

Version 12.1.0.2.0

Instant Client Package - Basic: All files required to run OCI, OCCI, and JDBC-OCI applications
[instantclient-basic-nt-12.1.0.2.0.zip](#) (67,410,370 bytes) (cksum - 530881682)

*Instant Client Package - Basic Lite: Smaller version of the Basic, with only English error messages and Unicode, ASCII, and Western European character set support
[instantclient-basclite-nt-12.1.0.2.0.zip](#) (33,851,306 bytes) (cksum - 3971707995)

*Instant Client Package - JDBC Supplement: Additional support for XA, Internationalization, and RowSet operations under JDBC
[instantclient-jdbc-nt-12.1.0.2.0.zip](#) (1,573,642 bytes) (cksum - 2509633653)

*Instant Client Package - SQL*Plus: Additional libraries and executable for running SQL*Plus with Instant Client
[instantclient-sqlplus-nt-12.1.0.2.0.zip](#) (805,513 bytes) (cksum - 2478532846)

*Instant Client Package - SDK: Additional header files and an example makefile for developing Oracle applications with Instant Client
[instantclient-sdk-nt-12.1.0.2.0.zip](#) (1,951,770 bytes) (cksum - 722480528)

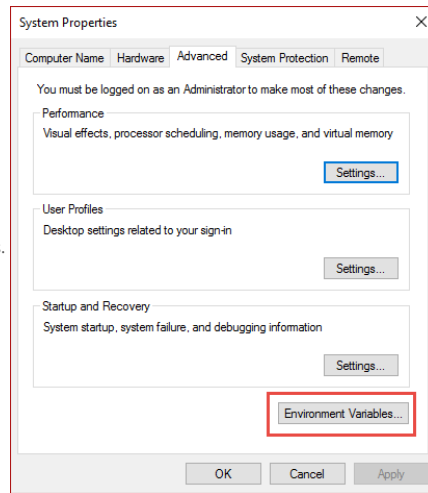
*Instant Client Package - ODBC: Additional libraries for enabling ODBC applications
[instantclient-odbc-nt-12.1.0.2.0.zip](#) (505,668 bytes) (cksum - 41700632)

*Instant Client Package - WRC: Workload Replay Client used to replay workload for RAT's DB Replay Feature
[instantclient-tools-nt-12.1.0.2.0.zip](#) (15,225 bytes) (cksum - 2411353204)

Download Instant Client 12.

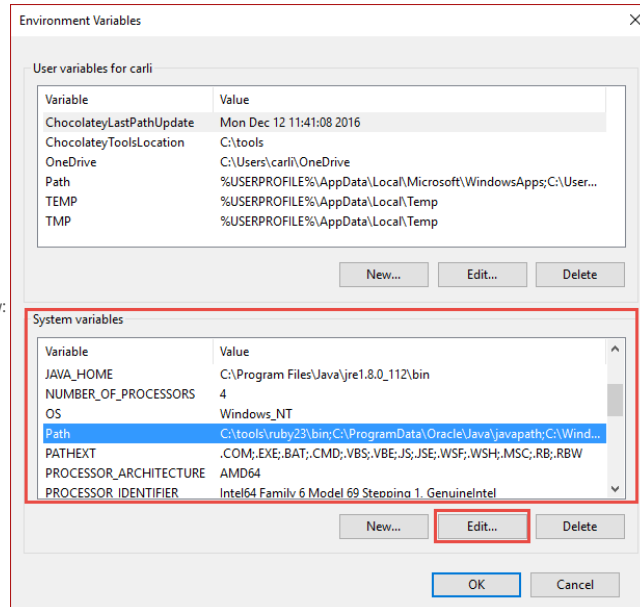
- xtract the **instant client basic** package from your computer (e.g.: C:\instantclient_12_1).
- e should put in the folder **PHP(NetMake/v9/components/php/)** the following DLLs:
 - o opy the dll **oraociei12.dll** (it is inside the folder of the instant client (e.g.: C:\instantclient_12_1)).
- dd to the system PATH the path of this folder:
 - o o to Control Panel> System> Advanced System Settings.

- o In the Advanced tab> Click Environment Variables.



Advanced Settings.

- o Add the path as shown below:



Adding client path.

- restart the web server

Oracle ODBC

Setting Up Oracle ODBC On Windows

The automatic installation already comes with the Oracle preconfigured, only needing to install the client so that you can do the connection.

*Enter Oracle's website (www.oracle.com) and download **Instant Client Basic** according to the architecture of your operating system (x64 or x86)

Note: PHP 7.0 is only compatible with instant client 11 or higher

Version 12.1.0.2.0

Instant Client Package - Basic: All files required to run OCI, OCCI, and JDBC-OCI applications
 Instantclient-basic-nt-12.1.0.2.0.zip (67,410,370 bytes) (cksum - 530881682)

*Instant Client Package - Basic Lite: Smaller version of the Basic, with only English error messages and Unicode, ASCII, and Western European character set support
 Instantclient-basicallyte-nt-12.1.0.2.0.zip (33,851,306 bytes) (cksum - 3971707995)

*Instant Client Package - JDBC Supplement: Additional support for XA, Internationalization, and RowSet operations under JDBC
 Instantclient-jdbc-nt-12.1.0.2.0.zip (1,573,642 bytes) (cksum - 2509633653)

*Instant Client Package - SQL*Plus: Additional libraries and executable for running SQL*Plus with Instant Client
 Instantclient-sqlplus-nt-12.1.0.2.0.zip (805,513 bytes) (cksum - 2478532846)

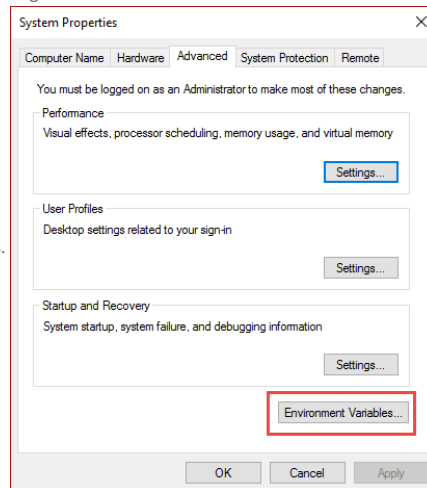
*Instant Client Package - SDK: Additional header files and an example makefile for developing Oracle applications with Instant Client
 Instantclient-sdk-nt-12.1.0.2.0.zip (1,951,770 bytes) (cksum - 722480528)

*Instant Client Package - ODBC: Additional libraries for enabling ODBC applications
 Instantclient-odbc-nt-12.1.0.2.0.zip (505,668 bytes) (cksum - 41700632)

*Instant Client Package - WRC: Workload Replay Client used to replay workload for RAT's DB Replay Feature
 Instantclient-tools-nt-12.1.0.2.0.zip (15,225 bytes) (cksum - 2411353204)

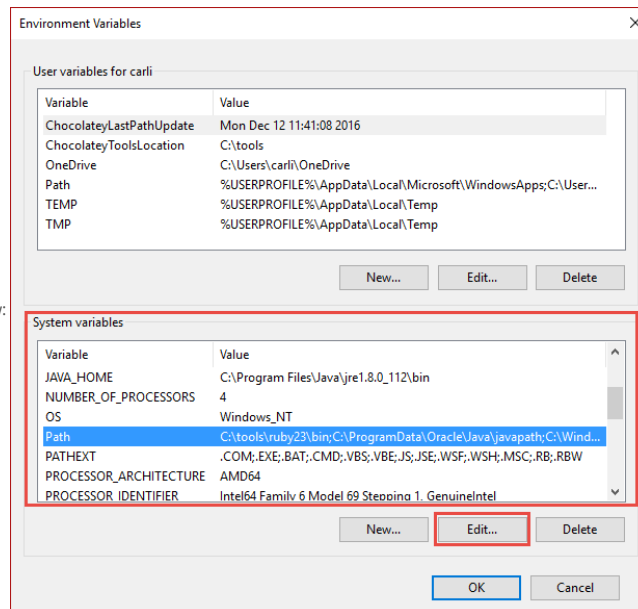
Download Instant Client 12.

- xtract the **instant client basic** package from your computer (e.g.: C:\instantclient_12_1).
- xtract the **instant client odbc** package inside the **instant client basic** folder (e.g.: C:\instantclient_12_1).
- e should put in the folder **PHP(NetMake/v9/components/php/)** the following DLLs:
 - opy the dll **oraociei12.dll** (it is inside the folder of the instant client (e.g.: C:\instantclient_12_1)).
- dd to the system PATH the path of this folder:
 - o to Control Panel> System> Advanced System Settings.



- n the Advanced tab> Click Environment Variables.

Advanced Settings.



- dd the path as shown below:

Adding client path.

- restart the web server

Setting Up Oracle 8 On Windows

The automatic installation already comes with the Oracle preconfigured, only needing to install the client so that you can do the connection.

*Enter Oracle's website (www.oracle.com) and download **Instant Client Basic** according to the architecture of your operating system (x64 or x86)

Note: PHP 7.0 is only compatible with instant client 11 or higher

Version 12.1.0.2.0

Instant Client Package - Basic: All files required to run OCI, OCCI, and JDBC-OCI applications
[instantclient-basic-nt-12.1.0.2.0.zip](#) (67,410,370 bytes) (cksum - 530881682)

*Instant Client Package - Basic Lite: Smaller version of the Basic, with only English error messages and Unicode, ASCII, and Western European character set support
[instantclient-basclite-nt-12.1.0.2.0.zip](#) (33,851,306 bytes) (cksum - 3971707995)

*Instant Client Package - JDBC Supplement: Additional support for XA, Internationalization, and RowSet operations under JDBC
[instantclient-jdbc-nt-12.1.0.2.0.zip](#) (1,573,642 bytes) (cksum - 2509633653)

*Instant Client Package - SQL*Plus: Additional libraries and executable for running SQL*Plus with Instant Client
[instantclient-sqlplus-nt-12.1.0.2.0.zip](#) (805,513 bytes) (cksum - 2478532846)

*Instant Client Package - SDK: Additional header files and an example makefile for developing Oracle applications with Instant Client
[instantclient-sdk-nt-12.1.0.2.0.zip](#) (1,951,770 bytes) (cksum - 722480528)

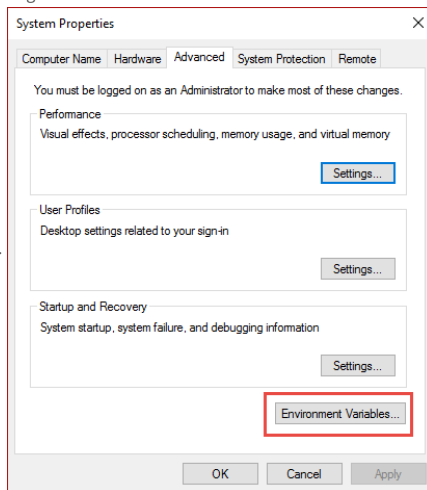
*Instant Client Package - ODBC: Additional libraries for enabling ODBC applications
[instantclient-odbc-nt-12.1.0.2.0.zip](#) (505,668 bytes) (cksum - 41700632)

*Instant Client Package - WRC: Workload Replay Client used to replay workload for RAT's DB Replay Feature
[instantclient-tools-nt-12.1.0.2.0.zip](#) (15,225 bytes) (cksum - 2411353204)

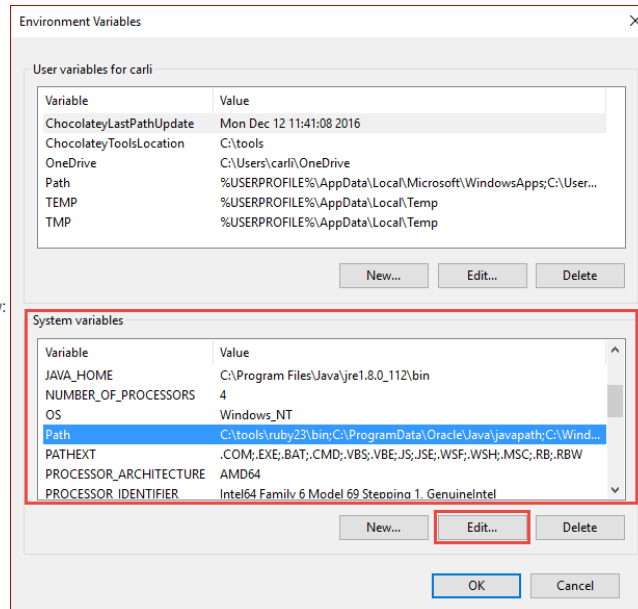
Download Instant Client 12.

- xtract the **instant client basic** package from your computer (e.g.: C:\instantclient_12_1).
- e should put in the folder **PHP(NetMake/v9/components/php/)** the following DLLs:
 - o py the dll **oraoci12.dll** (it is inside the folder of the instant client (e.g.: C:\instantclient_12_1)).
- dd to the system PATH the path of this folder:
 - o o to Control Panel> System> Advanced System Settings.

- o n the Advanced tab> Click Environment Variables.



Advanced Settings.



- o dd the path as shown below:

Adding client path.

- restart the web server

Oracle 7 or Lower

Setting Up Oracle 7 Or Lower On Windows

The automatic installation already comes with the Oracle preconfigured, only needing to install the client so that you can do the connection.

*Enter Oracle's website (www.oracle.com) and download **Instant Client Basic** according to the architecture of your operating system (x64 or x86)

Note: PHP 7.0 is only compatible with instant client 11 or higher

Version 12.1.0.2.0

Instant Client Package - Basic: All files required to run OCI, OCCI, and JDBC-OCI applications
[instantclient-basic-nt-12.1.0.2.0.zip](#) (67,410,370 bytes) (cksum - 530881682)

*Instant Client Package - Basic Lite: Smaller version of the Basic, with only English error messages and Unicode, ASCII, and Western European character set support
[instantclient-basclite-nt-12.1.0.2.0.zip](#) (33,851,306 bytes) (cksum - 3971707995)

*Instant Client Package - JDBC Supplement: Additional support for XA, Internationalization, and RowSet operations under JDBC
[instantclient-jdbc-nt-12.1.0.2.0.zip](#) (1,573,642 bytes) (cksum - 2509633653)

*Instant Client Package - SQL*Plus: Additional libraries and executable for running SQL*Plus with Instant Client
[instantclient-sqlplus-nt-12.1.0.2.0.zip](#) (805,513 bytes) (cksum - 2478532846)

*Instant Client Package - SDK: Additional header files and an example makefile for developing Oracle applications with Instant Client
[instantclient-sdk-nt-12.1.0.2.0.zip](#) (1,951,770 bytes) (cksum - 722480528)

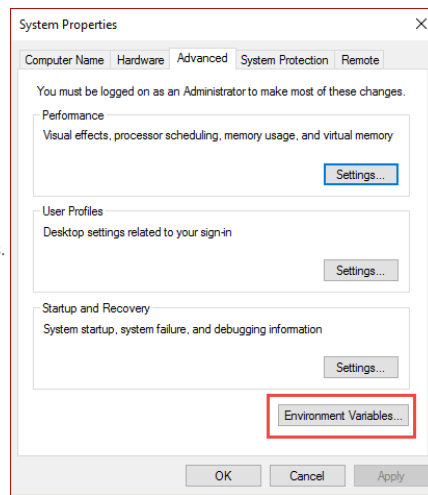
*Instant Client Package - ODBC: Additional libraries for enabling ODBC applications
[instantclient-odbc-nt-12.1.0.2.0.zip](#) (505,668 bytes) (cksum - 41700632)

*Instant Client Package - WRC: Workload Replay Client used to replay workload for RAT's DB Replay Feature
[instantclient-tools-nt-12.1.0.2.0.zip](#) (15,225 bytes) (cksum - 2411353204)

Download Instant Client 12.

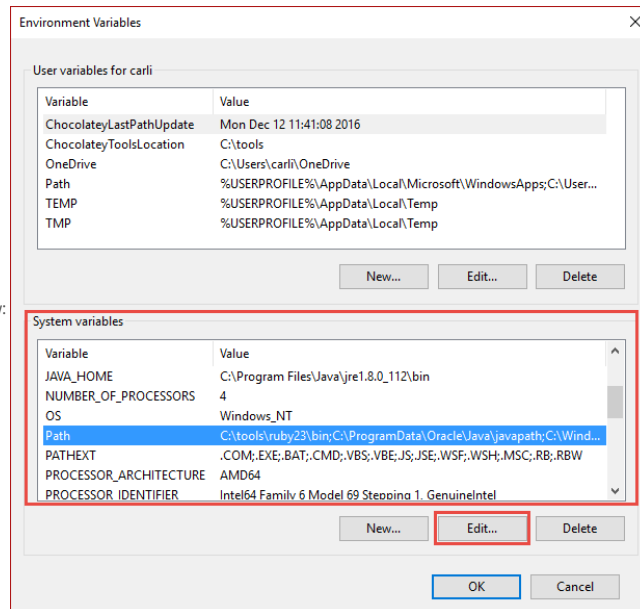
- xtract the **instant client basic** package from your computer (e.g.: C:\instantclient_12_1).
- e should put in the folder **PHP(NetMake/v9/components/php/)** the following DLLs:
 - o opy the dll **oraociei12.dll** (it is inside the folder of the instant client (e.g.: C:\instantclient_12_1)).
- dd to the system PATH the path of this folder:
 - o o to Control Panel> System> Advanced System Settings.

- o In the Advanced tab> Click Environment Variables.



Advanced Settings.

- o Add the path as shown below:



Adding client path.

- restart the web server

Creating a Connection with ScriptCase

- Access ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- After accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"















Creating a new connection

- After that, you will see a page with all the database connections that you can create. Select a "Oracle" connection

Create a connection

Select your database system and create a connection for your project.

 MySQL	 Oracle	 MSSQL Server	 Postgres	 DB2	 Informix
 MS Access	 SQLite	 SyBase	 InterBase	 Firebird	 ODBC

Selecting a Windows database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection Oracle

Enter your database connection details for Oracle

← Choose another connection

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver:

TNSNAME:

Username: Password:

Connecting with Windows database

- **onnection Name** : Defines the name of your new connection.
- **BMS Driver** : Allows you to select the drivers: Oracle 8.0.5 or Higher.
- **SNAME** : Insert the name or IP of the server where the Windows database is located with the Service Name created in the Windows installation.
- **ername** : Inform the user to access the Windows database.
- **assword** : Inform the password to access the Windows database.
- **est Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:

Create a connection Oracle

Enter your database connection details for Oracle

← Choose another connection

CONNECTION FILTER **ADVANCED**

client_encoding:

Decimal Separator:

Persistent Connection:

Use the schema before table name:

Advanced setup for the Windows database

- **lient_encoding** : Database Encoding.
- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that don't close when finishing a script.
- **se the schema before table name** - Allows the use of schemas before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER ADVANCED

Show

Tables

Views

System tables

Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No ▾

Test Connection
Save

Filtering the Windows database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.

The automatic installation already comes with the Oracle extension only needing to connect with the database.

If you are using your own previously configured environment, you will need to enable the Oracle extension in PHP.

Oracle PDO

Setting Up Oracle PDO On Linux

Due to a problem in the Oracle 64-bit extension, you can not use the connection with the automatic installation. To learn how to use Oracle with your 64bit Scriptcase [access the documentation](#).

The automatic installation already comes with the Oracle preconfigured, only needing to install the client so that you can do the connection.

- install the dependencies below:

UBUNTU/DEBIAN

```
- sudo apt-get update
- sudo apt-get install libaio1
- sudo apt-get install libncurses5
- sudo apt-get install alien
- sudo apt-get install gcc-multilib g++-multilib
- sudo apt-get install libpam0g
- sudo apt-get install unixodbc-dev unixodbc
```

CENTOS/RHEL

```
- sudo yum update
- sudo yum install libaio-devel
- sudo yum install libaio
- sudo yum install glibc
- sudo yum install compat-libstdc++-33
- sudo yum install glibc-devel
- sudo yum install libstdc++
- sudo yum install libstdc++
- sudo yum install pam
- sudo yum install ncurses-devel
- sudo yum install unixODBC
```

- download the instant client compatible with the architecture of your operating system, x64 or x86 bits
 - [Client basic Linux \(x86 \)](#)
 - [Client devel Linux \(x86 \)](#)
 - [Client basic Linux \(x64 \)](#)
 - [Client devel Linux \(x64 \)](#)

Installing Oracle Client

UBUNTU/DEBIAN

Architecture x86

```
sudo alien oracle-instantclient12.1-devel-12.1.0.2.0-1.i386.rpm
sudo alien oracle-instantclient12.1-basic-12.1.0.2.0-1.i386.rpm
sudo dpkg -i oracle-instantclient12.1-basic_12.1.0.2.0-2_i386.deb
sudo dpkg -i oracle-instantclient12.1-devel_12.1.0.2.0-2_i386.deb
```

Architecture x64

```
sudo alien oracle-instantclient12.1-devel-12.1.0.2.0-1.x86_64.rpm
sudo alien oracle-instantclient12.1-basic-12.1.0.2.0-1.x86_64.rpm
sudo dpkg -i oracle-instantclient12.1-basic_12.1.0.2.0-2_amd64.deb
sudo dpkg -i oracle-instantclient12.1-devel_12.1.0.2.0-2_amd64.deb
```

CENTOS/RHEL

Architecture x86

```
sudo yum install oracle-instantclient12.1-devel-12.1.0.2.0-1.i386.rpm
sudo yum install oracle-instantclient12.1-devel-12.1.0.2.0-1.x86_64.rpm
sudo yum install oracle-instantclient12.1-basic-12.1.0.2.0-1.i386.rpm
sudo yum install oracle-instantclient12.1-basic-12.1.0.2.0-1.x86_64.rpm
```

Architecture x64

```
sudo yum install oracle-instantclient12.1-devel-12.1.0.2.0-1.x86_64.rpm
sudo yum install oracle-instantclient12.1-basic-12.1.0.2.0-1.x86_64.rpm
```

Restart the Apache Server:

```
sudo /etc/init.d/apachesc9 restart
```

Oracle 8.0.5 or Higher

Setting Up Oracle 8.0.5 Or Higher On Linux

Due to a problem in the Oracle 64-bit extension, you can not use the connection with the automatic installation. To learn how to use Oracle with your 64bit Scriptcase [access the documentation](#).

The automatic installation already comes with the Oracle preconfigured, only needing to install the client so that you can do the connection.

- install the dependencies below:

UBUNTU/DEBIAN

```
- sudo apt-get update
- sudo apt-get install libaio1
- sudo apt-get install libncurses5
- sudo apt-get install alien
- sudo apt-get install gcc-multilib g++-multilib
- sudo apt-get install libpam0g
- sudo apt-get install unixodbc-dev unixodbc
```

CENTOS/RHEL

```
- sudo yum update
- sudo yum install libaio-devel
- sudo yum install libaio
- sudo yum install glibc
- sudo yum install compat-libstdc++-33
- sudo yum install glibc-devel
- sudo yum install libstdc++
- sudo yum install libstdc++
- sudo yum install pam
- sudo yum install ncurses-devel
- sudo yum install unixODBC
```

- download the instant client compatible with the architecture of your operating system, x64 or x86 bits
 - [Client basic Linux \(x86 \)](#)
 - [Client devel Linux \(x86 \)](#)
 - [Client basic Linux \(x64 \)](#)
 - [Client devel Linux \(x64 \)](#)

Installing Oracle Client

UBUNTU/DEBIAN

Architecture x86

```
sudo alien oracle-instantclient12.1-devel-12.1.0.2.0-1.i386.rpm
sudo alien oracle-instantclient12.1-basic-12.1.0.2.0-1.i386.rpm
sudo dpkg -i oracle-instantclient12.1-basic_12.1.0.2.0-2_i386.deb
sudo dpkg -i oracle-instantclient12.1-devel_12.1.0.2.0-2_i386.deb
```

Architecture x64

```
sudo alien oracle-instantclient12.1-devel-12.1.0.2.0-1.x86_64.rpm
sudo alien oracle-instantclient12.1-basic-12.1.0.2.0-1.x86_64.rpm
sudo dpkg -i oracle-instantclient12.1-basic_12.1.0.2.0-2_amd64.deb
sudo dpkg -i oracle-instantclient12.1-devel_12.1.0.2.0-2_amd64.deb
```

CENTOS/RHEL

Architecture x86

```
sudo yum install oracle-instantclient12.1-devel-12.1.0.2.0-1.i386.rpm
sudo yum install oracle-instantclient12.1-devel-12.1.0.2.0-1.x86_64.rpm
sudo yum install oracle-instantclient12.1-basic-12.1.0.2.0-1.i386.rpm
sudo yum install oracle-instantclient12.1-basic-12.1.0.2.0-1.x86_64.rpm
```

Architecture x64

Restart the Apache Server:

```
sudo /etc/init.d/apachesc9 restart
```

Oracle ODBC

Setting Up Oracle ODBC On Linux

Due to a problem in the Oracle 64-bit extension, you can not use the connection with the automatic installation. To learn how to use Oracle with your 64bit Scriptcase [access the documentation](#).

The automatic installation already comes with the Oracle preconfigured, only needing to install the client so that you can do the connection.

- install the dependencies below:

UBUNTU/DEBIAN

```
- sudo apt-get update
- sudo apt-get install libaio1
- sudo apt-get install libncurses5
- sudo apt-get install alien
- sudo apt-get install gcc-multilib g++-multilib
- sudo apt-get install libpam0g
- sudo apt-get install unixodbc-dev unixodbc
```

CENTOS/RHEL

```
- sudo yum update
- sudo yum install libaio-devel
- sudo yum install libaio
- sudo yum install glibc
- sudo yum install compat-libstdc++-33
- sudo yum install glibc-devel
- sudo yum install libstdc++
- sudo yum install libstdc++
- sudo yum install pam
- sudo yum install ncurses-devel
- sudo yum install unixODBC
```

- download the instant client compatible with the architecture of your operating system, x64 or x86 bits
 - [Client basic Linux \(x86 \)](#)
 - [Client devel Linux \(x86 \)](#)

- [Client devel Linux \(x86 \)](#)
- [Client basic Linux \(x64 \)](#)
- [Client devel Linux \(x64 \)](#)
- [Client devel Linux \(x64 \)](#)

Installing Oracle Client

UBUNTU/DEBIAN

```
Architecture x86
sudo alien oracle-instantclient12.1-devel-12.1.0.2.0-1.i386.rpm
sudo alien oracle-instantclient12.1-basic-12.1.0.2.0-1.i386.rpm
sudo alien oracle-instantclient12.1-odbc-12.1.0.2.0-1.i386.rpm
sudo dpkg -i oracle-instantclient12.1-basic_12.1.0.2.0-2_i386.deb
sudo dpkg -i oracle-instantclient12.1-devel_12.1.0.2.0-2_i386.deb
sudo dpkg -i oracle-instantclient12.1-odbc-12.1.0.2.0-2_i386.deb

Architecture x64
sudo alien oracle-instantclient12.1-devel-12.1.0.2.0-1.x86_64.rpm
sudo alien oracle-instantclient12.1-basic-12.1.0.2.0-1.x86_64.rpm
sudo alien oracle-instantclient12.1-odbc-12.1.0.2.0-1.x86_64.rpm
sudo dpkg -i oracle-instantclient12.1-basic_12.1.0.2.0-2_amd64.deb
sudo dpkg -i oracle-instantclient12.1-devel_12.1.0.2.0-2_amd64.deb
sudo dpkg -i oracle-instantclient12.1-odbc-12.1.0.2.0-2_amd64.deb
```

CENTOS/RHEL

```
Architecture x86
sudo yum install oracle-instantclient12.1-devel-12.1.0.2.0-1.i386.rpm
sudo yum install oracle-instantclient12.1-basic-12.1.0.2.0-1.i386.rpm
sudo yum install oracle-instantclient12.1-odbc-12.1.0.2.0-1.i386.rpm

Architecture x64
sudo yum install oracle-instantclient12.1-devel-12.1.0.2.0-1.x86_64.rpm
sudo yum install oracle-instantclient12.1-basic-12.1.0.2.0-1.x86_64.rpm
sudo yum install oracle-instantclient12.1-odbc-12.1.0.2.0-1.x86_64.rpm
```

Restart the Apache Server:

```
sudo /etc/init.d/apachesc9 restart
```

Oracle 8

Setting Up Oracle 8 On Linux

Due to a problem in the Oracle 64-bit extension, you can not use the connection with the automatic installation. To learn how to use Oracle with your 64bit Scriptcase [access the documentation](#).

The automatic installation already comes with the Oracle preconfigured, only needing to install the client so that you can do the connection.

- install the dependencies below:

UBUNTU/DEBIAN

```
- sudo apt-get update
- sudo apt-get install libaio1
- sudo apt-get install libncurses5
- sudo apt-get install alien
- sudo apt-get install gcc-multilib g++-multilib
- sudo apt-get install libpam0g
- sudo apt-get install unixodbc-dev unixodbc
```

CENTOS/RHEL

```
- sudo yum update
- sudo yum install libaio-devel
- sudo yum install libaio
- sudo yum install glibc
- sudo yum install compat-libstdc++-33
- sudo yum install glibc-devel
- sudo yum install libstdc++
- sudo yum install libstdc++
- sudo yum install pam
- sudo yum install ncurses-devel
- sudo yum install unixODBC
```

- download the instant client compatible with the architecture of your operating system, x64 or x86 bits
 - [Client basic Linux \(x86 \)](#)
 - [Client devel Linux \(x86 \)](#)
 - [Client basic Linux \(x64 \)](#)
 - [Client devel Linux \(x64 \)](#)

Installing Oracle Client

UBUNTU/DEBIAN

```
Architecture x86
sudo alien oracle-instantclient12.1-devel-12.1.0.2.0-1.i386.rpm
sudo alien oracle-instantclient12.1-basic-12.1.0.2.0-1.i386.rpm
sudo dpkg -i oracle-instantclient12.1-basic_12.1.0.2.0-2_i386.deb
sudo dpkg -i oracle-instantclient12.1-devel_12.1.0.2.0-2_i386.deb

Architecture x64
sudo alien oracle-instantclient12.1-devel-12.1.0.2.0-1.x86_64.rpm
sudo alien oracle-instantclient12.1-basic-12.1.0.2.0-1.x86_64.rpm
sudo dpkg -i oracle-instantclient12.1-basic_12.1.0.2.0-2_amd64.deb
sudo dpkg -i oracle-instantclient12.1-devel_12.1.0.2.0-2_amd64.deb
```

CENTOS/RHEL

```
Architecture x86
sudo yum install oracle-instantclient12.1-devel-12.1.0.2.0-1.i386.rpm

Architecture x64
sudo yum install oracle-instantclient12.1-devel-12.1.0.2.0-1.x86_64.rpm
```

CENTOSRHEL `sudo yum install oracle-instantclient12.1-basic-12.1.0.2.0-1.i386.rpm`

Restart the Apache Server:

`sudo /etc/init.d/apachesc9 restart`

Oracle 7 or Lower

Setting Up Oracle 7 Or Lower On Linux

Due to a problem in the Oracle 64-bit extension, you can not use the connection with the automatic installation. To learn how to use Oracle with your 64bit Scriptcase [access the documentation](#).

The automatic installation already comes with the Oracle preconfigured, only needing to install the client so that you can do the connection.

- install the dependencies below:

UBUNTUDEBIAN

```
- sudo apt-get update
- sudo apt-get install libaio1
- sudo apt-get install libncurses5
- sudo apt-get install alien
- sudo apt-get install gcc-multilib g++-multilib
- sudo apt-get install libpam0g
- sudo apt-get install unixodbc-dev unixodbc
```

CENTOSRHEL

```
- sudo yum update
- sudo yum install libaio-devel
- sudo yum install libaio
- sudo yum install glibc
- sudo yum install compat-libstdc++-33
- sudo yum install glibc-devel
- sudo yum install libstdc++
- sudo yum install libstdc++
- sudo yum install pam
- sudo yum install ncurses-devel
- sudo yum install unixODBC
```

- download the instant client compatible with the architecture of your operating system, x64 or x86 bits
 - [Client basic Linux \(x86 \)](#)
 - [Client devel Linux \(x86 \)](#)
 - [Client basic Linux \(x64 \)](#)
 - [Client devel Linux \(x64 \)](#)

Installing Oracle Client

UBUNTUDEBIAN

```
Architecture x86 Architecture x64
sudo alien oracle-instantclient12.1-devel-12.1.0.2.0-1.i386.rpm sudo alien oracle-instantclient12.1-devel-12.1.0.2.0-1.x86_64.rpm
sudo alien oracle-instantclient12.1-basic-12.1.0.2.0-1.i386.rpm sudo alien oracle-instantclient12.1-basic-12.1.0.2.0-1.x86_64.rpm
sudo dpkg -i oracle-instantclient12.1-basic_12.1.0.2.0-2_i386.deb sudo dpkg -i oracle-instantclient12.1-basic_12.1.0.2.0-2_amd64.deb
sudo dpkg -i oracle-instantclient12.1-devel_12.1.0.2.0-2_i386.deb sudo dpkg -i oracle-instantclient12.1-devel_12.1.0.2.0-2_amd64.deb
```

CENTOSRHEL

```
Architecture x86 Architecture x64
sudo yum install oracle-instantclient12.1-devel-12.1.0.2.0-1.i386.rpm sudo yum install oracle-instantclient12.1-devel-12.1.0.2.0-1.x86_64.rpm
sudo yum install oracle-instantclient12.1-basic-12.1.0.2.0-1.i386.rpm sudo yum install oracle-instantclient12.1-basic-12.1.0.2.0-1.x86_64.rpm
```

Restart the Apache Server:

`sudo /etc/init.d/apachesc9 restart`

Creating a Connection with ScriptCase

- access ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- after accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"















Creating a new connection

- after that, you will see a page with all the database connections that you can create. Select a "Oracle" connection

Create a connection

Select your database system and create a connection for your project.

 MySQL	 Oracle	 Microsoft SQL Server	 PostgreSQL	 DB2	 Informix
 MS Access	 SQLite	 SyBase	 InterBase	 Firebird	 ODBC

Selecting a Linux database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection Oracle

Enter your database connection details for Oracle ← Choose another connection

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver:

TNSNAME:

Username: Password:

Test Connection Save

Connecting with Linux database

- **onnection Name** : Defines the name of your new connection.
- **BMS Driver** : Allows you to select the drivers: Oracle 8.0.5 or Higher.
- **SNAME** : Insert the name or IP of the server where the Linux database is located with the Service Name created in the Linux installation.
- **sername** : Inform the user to access the Linux database.
- **assword** : Inform the password to access the Linux database.
- **est Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:

Create a connection Oracle

Enter your database connection details for Oracle ← Choose another connection

CONNECTION FILTER ADVANCED

client_encoding:

Decimal Separator:

Persistent Connection:

Use the schema before table name:

Test Connection Save

Advanced setup for the Linux database

- **lient_encoding** : Database Encoding.
- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that don't close when finishing a script.
- **se the schema before table name** - Allows the use of schemas before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER ADVANCED

Show

Tables

Views

System tables

Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No ▾

Test Connection
Save

Filtering the Linux database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.

The automatic installation already comes with the Oracle extension only needing to connect with the database.

If you are using your own previously configured environment, you will need to enable the Oracle extension in PHP.

Oracle PDO

Setting Up Oracle PDO On MAC

The automatic installation already comes with the Oracle preconfigured, only needing to install the client so that you can do the connection.

- download the Oracle Instant Client
 - [Client Oracle 12 Mac OS X](#)
- create the folder `/usr/local/instantclient/12.1.0.2/`
- extract all content from the downloaded file into `/usr/local/instantclient/12.1.0.2/`
- restart the web server.

Oracle 8.0.5 or Higher

Setting Up Oracle 8.0.5 Or Higher On MAC

The automatic installation already comes with the Oracle preconfigured, only needing to install the client so that you can do the connection.

- download the Oracle Instant Client
 - [Client Oracle 12 Mac OS X](#)
- create the folder `/usr/local/instantclient/12.1.0.2/`
- extract all content from the downloaded file into `/usr/local/instantclient/12.1.0.2/`
- restart the web server.

Oracle ODBC

Setting Up Oracle ODBC On MAC

The automatic installation already comes with the Oracle preconfigured, only needing to install the client so that you can do the connection.

- download the Oracle Instant Client
 - [Client Oracle 12 Mac OS X](#)
- create the folder `/usr/local/instantclient/12.1.0.2/`
- extract all content from the downloaded file into `/usr/local/instantclient/12.1.0.2/`
- restart the web server.

Oracle 8

Setting Up Oracle 8 On MAC

The automatic installation already comes with the Oracle preconfigured, only needing to install the client so that you can do the connection.

- download the Oracle Instant Client
 - [Client Oracle 12 Mac OS X](#)
- create the folder `/usr/local/instantclient/12.1.0.2/`

- extract all content from the downloaded file into `/usr/local/instanclient/12.1.0.2/`
- restart the web server.

Oracle 7 or Lower

Setting Up Oracle 7 Or Lower On MAC

The automatic installation already comes with the Oracle preconfigured, only needing to install the client so that you can do the connection.

- download the Oracle Instant Client
 - [Client Oracle 12 Mac OS X](#)
- create the folder `/usr/local/instanclient/12.1.0.2/`
- extract all content from the downloaded file into `/usr/local/instanclient/12.1.0.2/`
- restart the web server.

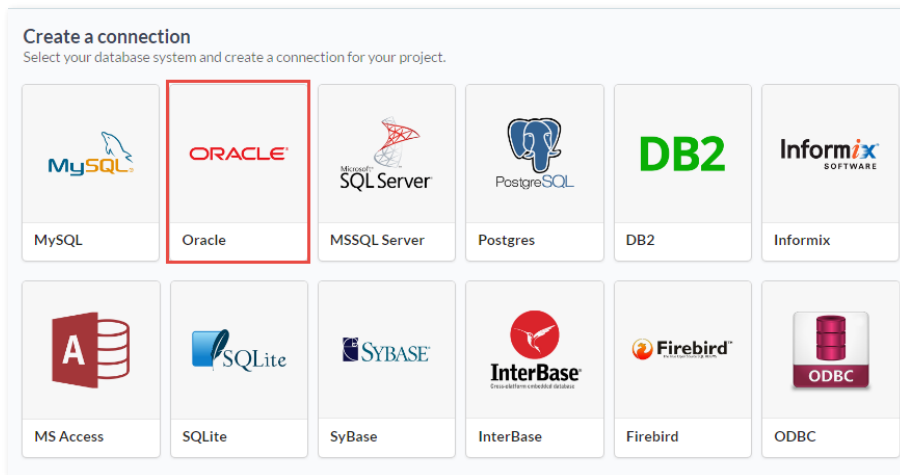
Creating a Connection with ScriptCase

- access ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- after accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- after that, you will see a page with all the database connections that you can create. Select a "Oracle" connection



Selecting a Mac OS X database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection Oracle
Enter your database connection details for Oracle

← Choose another connection

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver:

TNSNAME:

Username: Password:

Connecting with Mac OS X database

- **onnection Name** : Defines the name of your new connection.
- **BMS Driver** : Allows you to select the drivers: Oracle 8.0.5 or Higher.
- **SNAME** : Insert the name or IP of the server where the Mac OS X database is located with the Service Name created in the Mac OS X installation.

- **sername** : Inform the user to access the Mac OS X database.
- **assword** : Inform the password to access the Mac OS X database.
- **est Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:

Create a connection Oracle
Enter your database connection details for Oracle

← Choose another connection

CONNECTION FILTER **ADVANCED**

client_encoding

Decimal Separator

Persistent Connection

Use the schema before table name

Test Connection Save

Advanced setup for the Mac OS X database

- **lient_encoding** : Database Encoding.
- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that don't close when finishing a script.
- **se the schema before table name** - Allows the use of schemas before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION **FILTER** ADVANCED

Show

Tables
 Views
 System tables
 Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No

Test Connection Save

Filtering the Mac OS X database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.



The automatic installation already comes with the PostgreSQL extension only needing to connect with the database.

If you are using your own previously configured environment, you will need to enable the PostgreSQL extension in PHP.

PostgreSQL 7 or Higher

Setting Up PostgreSQL 7 Or Higher On Windows

The automatic installation already comes with the PostgreSQL only needing to connect with the database with ScriptCase.

PostgreSQL PDO

Setting Up PostgreSQL PDO On Windows

The automatic installation already comes with the PostgreSQL only needing to connect with the database with ScriptCase.

PostgreSQL 6.4 or Higher

Setting Up PostgreSQL 6.4 Or Higher On Windows

The automatic installation already comes with the PostgreSQL only needing to connect with the database with ScriptCase.

PostgreSQL 6.3 or Lower

Setting Up PostgreSQL 6.3 Or Lower On Windows

The automatic installation already comes with the PostgreSQL only needing to connect with the database with ScriptCase.

Creating a Connection with ScriptCase

- Access ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- After accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- After that, you will see a page with all the database connections that you can create. Select a "PostgreSQL" connection

Create a connection

Select your database system and create a connection for your project.

The screenshot shows a grid of database system icons. The PostgreSQL icon, featuring a blue elephant, is highlighted with a red border. Other visible options include MySQL, Oracle, MSSQL Server, DB2, Informix, MS Access, SQLite, SyBase, InterBase, Firebird, and ODBC.

Selecting a Windows database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection Postgres

Enter your database connection details for Postgres

← Choose another connection

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver: PostgreSQL PDO

Server/Host (Name or IP): Port:

Username: Password:

Database Name:

Connecting with Windows database

- **onnection Name** : Defines the name of your new connection.
- **BMS Driver** : Allows you to select the drivers: PostgreSQL PDO.
- **erver/Host (Name or IP)** : Insert the name or IP of the server where the Windows database is located.
- **ort** : Define the port used for the connection. By default it is 5432.
- **atabase Name** : Select the database.
- **ername** : Inform the user to access the Windows database.
- **assword** : Inform the password to access the Windows database.
- **est Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:

Create a connection Postgres

Enter your database connection details for Postgres

← Choose another connection

CONNECTION FILTER ADVANCED

client_encoding:

Decimal Separator:

Persistent Connection:

Use the schema before table name:

Advanced setup for the Windows database

- **lient_encoding** : Database Encoding.
- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that do not close when finishing a script.

- **se the schema before table name** - Allows the use of schemas before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:

The screenshot shows a 'FILTER' tab with two main sections: 'Show' and 'Searches'. The 'Show' section contains four checkboxes: 'Tables' (checked), 'Views' (checked), 'System tables' (unchecked), and 'Procedures' (unchecked). The 'Searches' section contains three input fields: 'Tables' (empty), 'Owner' (empty), and 'Show' (set to 'No'). At the bottom of the tab are two buttons: 'Test Connection' (green) and 'Save' (blue).

Filtering the Windows database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.

The automatic installation already comes with the PostgreSQL extension only needing to connect with the database.

If you are using your own previously configured environment, you will need to enable the PostgreSQL extension in PHP.

PostgreSQL 7 or Higher

Setting Up PostgreSQL 7 Or Higher On Linux

The automatic installation already comes with the PostgreSQL only needing to connect with the database with ScriptCase.

PostgreSQL PDO

Setting Up PostgreSQL PDO On Linux

The automatic installation already comes with the PostgreSQL only needing to connect with the database with ScriptCase.

PostgreSQL 6.4 or Higher

Setting Up PostgreSQL 6.4 Or Higher On Linux

The automatic installation already comes with the PostgreSQL only needing to connect with the database with ScriptCase.

PostgreSQL 6.3 or Lower

Setting Up PostgreSQL 6.3 Or Lower On Linux

The automatic installation already comes with the PostgreSQL only needing to connect with the database with ScriptCase.

Creating a Connection with ScriptCase

- Access ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- After accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- After that, you will see a page with all the database connections that you can create. Select a "PostgreSQL" connection

Create a connection

Select your database system and create a connection for your project.

The interface displays a grid of database system logos and names. The PostgreSQL option is highlighted with a red border. The options are:

- MySQL
- Oracle
- MSSQL Server
- Postgres (highlighted)
- DB2
- Informix
- MS Access
- SQLite
- SyBase
- InterBase
- Firebird
- ODBC

Selecting a Linux database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection Postgres

Enter your database connection details for Postgres

← Choose another connection

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver: PostgreSQL PDO

Server/Host (Name or IP): Port:

Username: Password:

Database Name:

Connecting with Linux database

- **onnection Name** : Defines the name of your new connection.
- **BMS Driver** : Allows you to select the drivers: PostgreSQL PDO.
- **erver/Host (Name or IP)** : Insert the name or IP of the server where the Linux database is located.
- **ort** : Define the port used for the connection. By default it is 5432.
- **atabase Name** : Select the database.
- **ername** : Inform the user to access the Linux database.
- **assword** : Inform the password to access the Linux database.
- **est Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:

Create a connection Postgres

Enter your database connection details for Postgres

← Choose another connection

CONNECTION FILTER ADVANCED

client_encoding:

Decimal Separator:

Persistent Connection:

Use the schema before table name:

Advanced setup for the Linux database

- **lient_encoding** : Database Encoding.
- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that do not close when finishing a script.

- **se the schema before table name** - Allows the use of schemas before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:

The screenshot shows a 'FILTER' tab with two main sections: 'Show' and 'Searches'.
 In the 'Show' section, there are four checkboxes: 'Tables' (checked), 'Views' (checked), 'System tables' (unchecked), and 'Procedures' (unchecked).
 In the 'Searches' section, there are three input fields: 'Tables' (empty), 'Owner' (empty), and 'Show' (a dropdown menu currently showing 'No').
 At the bottom of the tab, there are two buttons: 'Test Connection' (green) and 'Save' (blue).

Filtering the Linux database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.

The automatic installation already comes with the PostgreSQL extension only needing to connect with the database.

If you are using your own previously configured environment, you will need to enable the PostgreSQL extension in PHP.

PostgreSQL 7 or Higher

Setting Up PostgreSQL 7 Or Higher On MAC

The automatic installation already comes with the PostgreSQL only needing to connect with the database with ScriptCase.

PostgreSQL PDO

Setting Up PostgreSQL PDO On MAC

The automatic installation already comes with the PostgreSQL only needing to connect with the database with ScriptCase.

PostgreSQL 6.4 or Higher

Setting Up PostgreSQL 6.4 Or Higher On MAC

The automatic installation already comes with the PostgreSQL only needing to connect with the database with ScriptCase.

PostgreSQL 6.3 or Lower

Setting Up PostgreSQL 6.3 Or Lower On MAC

The automatic installation already comes with the PostgreSQL only needing to connect with the database with ScriptCase.

Creating a Connection with ScriptCase

- Access ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- After accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- After that, you will see a page with all the database connections that you can create. Select a "PostgreSQL" connection

Create a connection

Select your database system and create a connection for your project.

The screenshot shows a grid of database system icons. The PostgreSQL icon, featuring a blue elephant, is highlighted with a red border. Other visible options include MySQL, Oracle, MSSQL Server, DB2, Informix, MS Access, SQLite, SyBase, InterBase, Firebird, and ODBC.

Selecting a Mac OS X database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection Postgres

Enter your database connection details for Postgres

← Choose another connection

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver: PostgreSQL PDO

Server/Host (Name or IP): Port:

Username: Password:

Database Name:

Connecting with Mac OS X database

- **onnection Name** : Defines the name of your new connection.
- **BMS Driver** : Allows you to select the drivers: PostgreSQL PDO.
- **erver/Host (Name or IP)** : Insert the name or IP of the server where the Mac OS X database is located.
- **ort** : Define the port used for the connection. By default it is 5432.
- **atabase Name** : Select the database.
- **ername** : Inform the user to access the Mac OS X database.
- **assword** : Inform the password to access the Mac OS X database.
- **est Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:

Create a connection Postgres

Enter your database connection details for Postgres

← Choose another connection

CONNECTION FILTER ADVANCED

client_encoding:

Decimal Separator:

Persistent Connection:

Use the schema before table name:

Advanced setup for the Mac OS X database

- **lient_encoding** : Database Encoding.
- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that do not close when finishing a script.

- **se the schema before table name** - Allows the use of schemas before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:

The screenshot shows a 'FILTER' tab with two main sections: 'Show' and 'Searches'. The 'Show' section contains four checkboxes: 'Tables' (checked), 'Views' (checked), 'System tables' (unchecked), and 'Procedures' (unchecked). The 'Searches' section contains three input fields: 'Tables', 'Owner', and 'Show'. The 'Show' dropdown menu is currently set to 'No'. At the bottom of the tab are two buttons: 'Test Connection' (green) and 'Save' (blue).

Filtering the Mac OS X database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.



See below the compatibility table of the drivers available in Scriptcase with Windows.

Compatibility Table

	Server 2016	Server 2012 /R2	Windows 10	Windows 8 / 8.1	Server 2008 SP2 / R2 SP1	Windows 7 SP1	Windows Vista SP2
Native SRV PDO	✓	✓	✓	✓	✓	✓	
Native SRV	✓	✓	✓	✓	✓	✓	
ODBC	✓	✓	✓	✓	✓	✓	✓

if you need to use [Database Identifiers](#) in or tables, we recommend using the double quotation marks(") instead of the brackets([]) because of the conflict with the [Global Variables syntax](#) of Scriptcase. Using the brackets as database identifiers may lead to problems in the Scriptcase applications.

MSSQL Server Native SRV

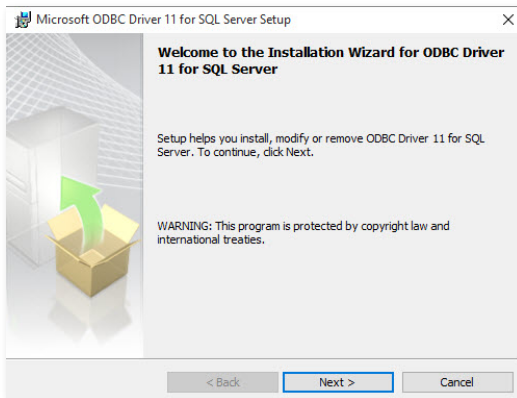
Setting Up MSSQL Server Native SRV On Windows

The automatic installation already comes with the MSSQL Server previously configured, only needing to install the Microsoft ODBC Driver 11 for SQL Server so that you can do the connection.

Supported Operating Systems: Windows Server (from the 2008 SP2 version) and Windows (from the 7 SP1 version).

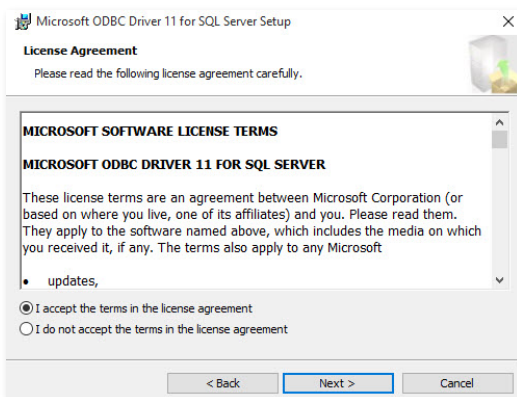
Installing Microsoft ODBC Driver 11 For SQL Server

- download the Microsoft ODBC Driver 11 for SQL Server according to the operating system architecture:
 - [ODBC Driver 11 64 bits](#)
 - [ODBC Driver 11 32 bits](#)
- after downloading the file, run the installer and follow the steps below:



Installation wizard.

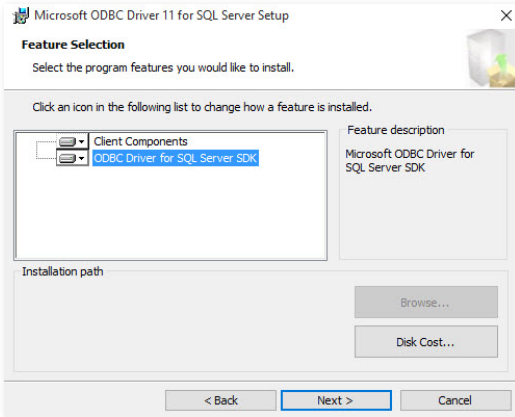
- accept the terms of the license agreement and click next.



Installation wizard.

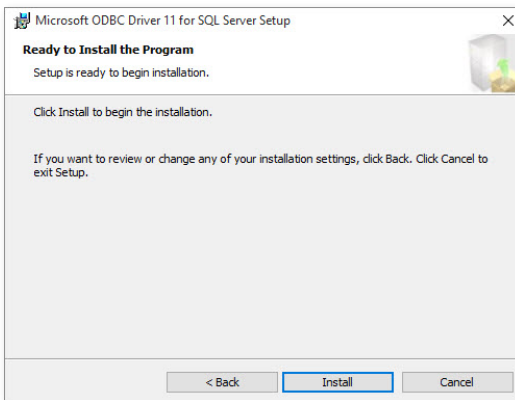
In this option leave the option "Client components" checked.

- check the "ODBC Driver for SQL Server SDK" option, because the components we are going to need are also in this option.



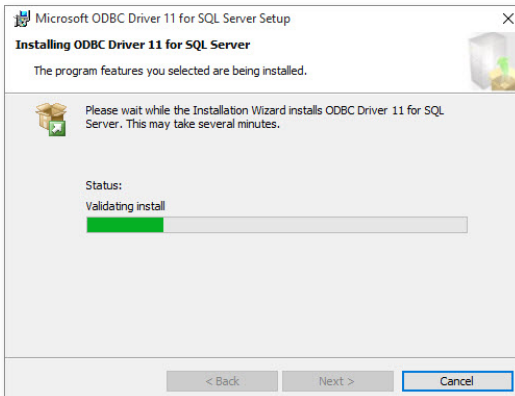
Installation wizard.

- click Install to begin the client installation.



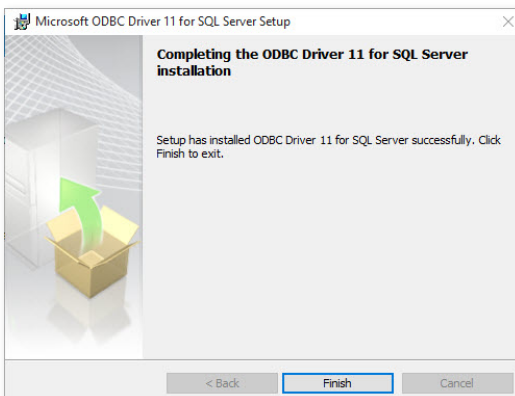
Installation wizard.

- the features of the program you selected are being installed.



Installation wizard.

- click Finish to exit.



MSSQL Server Native SRV PDO

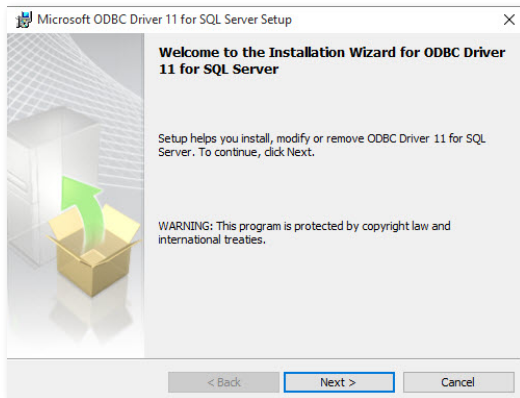
Setting Up MSSQL Server Native SRV On Windows

The automatic installation already comes with the MSSQL Server previously configured, only needing to install the Microsoft ODBC Driver 11 for SQL Server so that you can do the connection.

Supported Operating Systems: Windows Server (from the 2008 SP2 version) and Windows (from the 7 SP1 version).

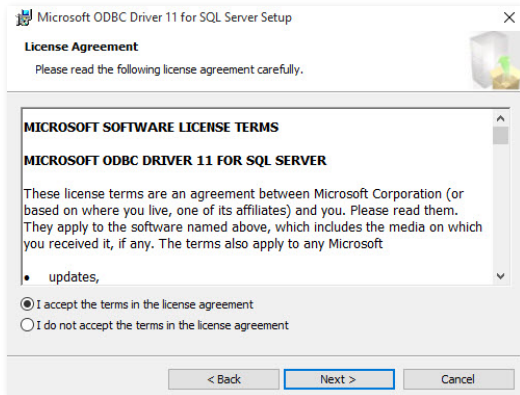
Installing Microsoft ODBC Driver 11 For SQL Server

- Download the Microsoft ODBC Driver 11 for SQL Server according to the operating system architecture:
 - [ODBC Driver 11 64 bits](#)
 - [ODBC Driver 11 32 bits](#)
- After downloading the file, run the installer and follow the steps below:



Installation wizard.

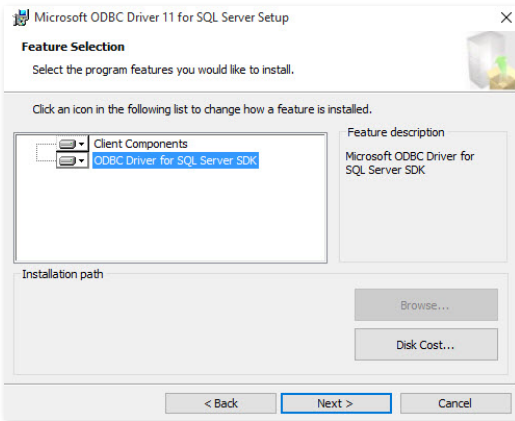
- Accept the terms of the license agreement and click next.



Installation wizard.

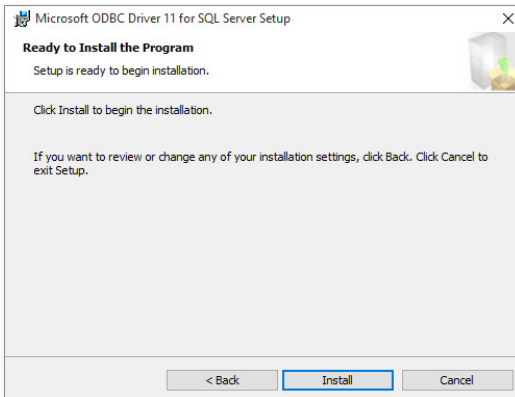
In this option leave the option "Client components" checked.

- Check the "ODBC Driver for SQL Server SDK" option, because the components we are going to need are also in this option.



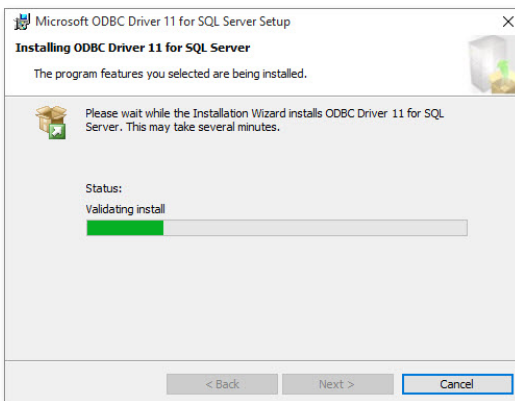
Installation wizard.

- lick Install to begin the client installation.



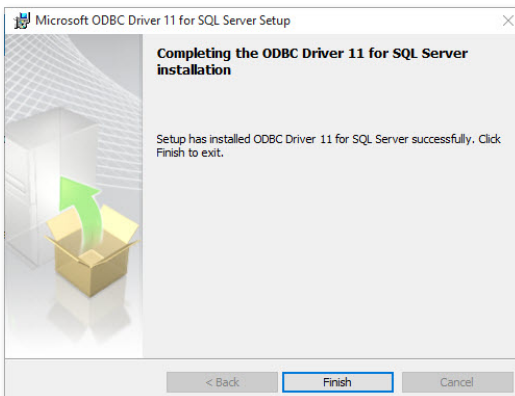
Installation wizard.

- he features of the program you selected are being installed.



Installation wizard.

- lick Finish to exit.



Installation wizard.

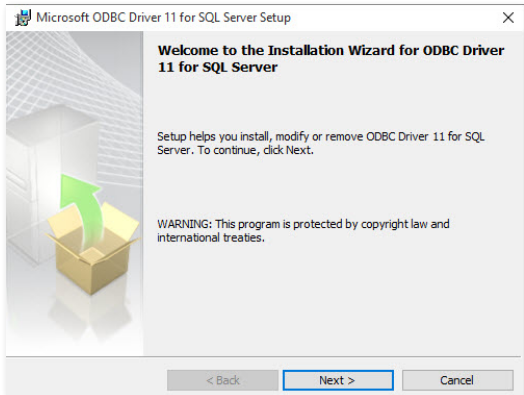
Setting Up MSSQL Server ODBC On Windows

The automatic installation already comes with the MSSQL Server previously configured, only needing to install the Microsoft ODBC Driver 11 for SQL Server so that you can do the connection.

Supported Operating Systems: Windows Server (from the 2008 SP2 version) and Windows (from the 7 SP1 version).

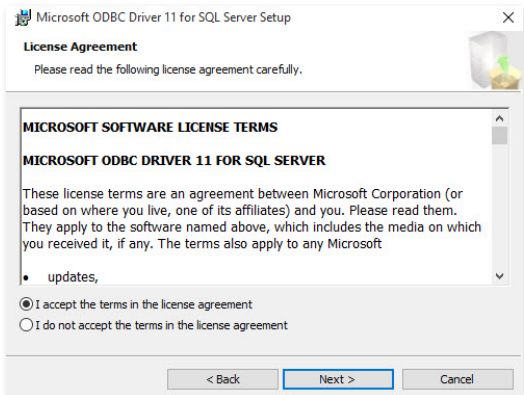
Installing Microsoft ODBC Driver 11 For SQL Server

- download the Microsoft ODBC Driver 11 for SQL Server according to the operating system architecture:
 - [ODBC Driver 11 64 bits](#)
 - [ODBC Driver 11 32 bits](#)
- after downloading the file, run the installer and follow the steps below:



Installation wizard.

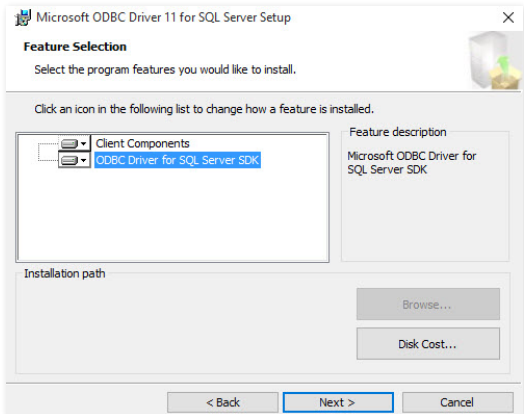
- accept the terms of the license agreement and click next.



Installation wizard.

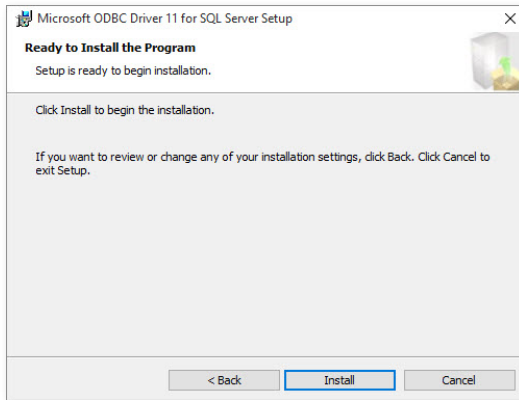
In this option leave the option "Client components" checked.

- check the "ODBC Driver for SQL Server SDK" option, because the components we are going to need are also in this option.



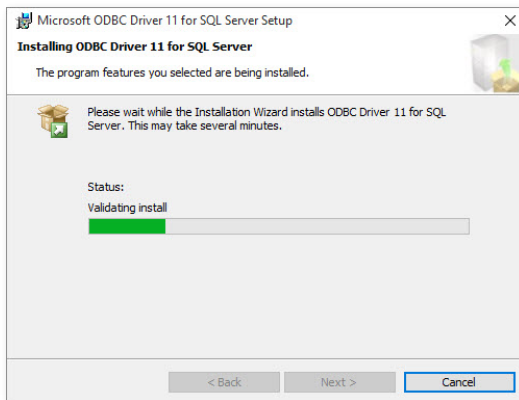
Installation wizard.

- lick Install to begin the client installation.



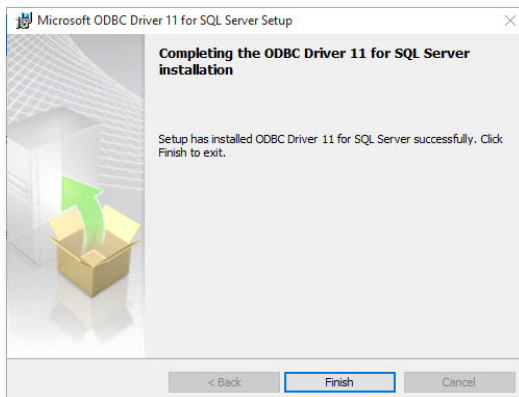
Installation wizard.

- he features of the program you selected are being installed.



Installation wizard.

- lick Finish to exit.



Installation wizard.

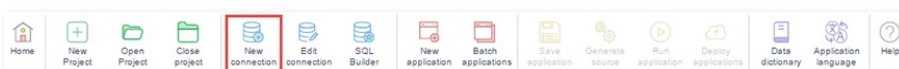
Configuring Data Source

- o connect to MS SQL Server ODBC you will need to create a system data source on the same Scriptcase server.
- o do so, go to: **Control Panel > System and Security > Administrative Tools > ODBC Data Sources**

After creating the system data source, you just need to create an MS SQL Server ODBC connection in the Scriptcase.

Creating a Connection with ScriptCase













- ccess ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- fter accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- After that, you will see a page with all the database connections that you can create. Select a "MSSQL Server" connection

Create a connection
Select your database system and create a connection for your project.

 MySQL	 Oracle	 MSSQL Server	 Postgres	 DB2	 Informix
 MS Access	 SQLite	 SyBase	 InterBase	 Firebird	 ODBC

Selecting a Windows database connection

- **new connection** : You will place the information of the connection to your database here:

Create a connection MSSQL Server
Enter your database connection details for MSSQL Server

← Choose another connection

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver:

Server/Host (Name or IP):

Username: Password:

Database Name:

Connecting with Windows database

- **Connection Name** : Defines the name of your new connection.
- **DBMS Driver** : Allows you to select the drivers: Native SRV PDO, Native SRV or ODBC.
- **Server/Host (Name or IP)** : Insert the name or IP\INSTANCE of the server where the Windows database is located.
- **Database Name** : Select the database.
- **Username** : Inform the user to access the Windows database.
- **Password** : Inform the password to access the Windows database.
- **Test Connection** : Displays a message of the status of the connection if successful or not.
- For more options on connecting, click on the **Advanced** tab:

Create a connection MSSQL Server
Enter your database connection details for MSSQL Server

← Choose another connection

CONNECTION FILTER **ADVANCED**

Decimal Separator:

Persistent Connection:

Use the schema before table name:

Advanced setup for the Windows database

- **Decimal Separator** - Select the separator type, between a dot or a comma.
- **Unicode Converter** - Select if there is going to be conversion of characters to Unicode.
- **Persistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **Use the schema before table name** - Allows the use of schema before the name of the tables.
- **Filter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER ADVANCED

Show

Tables

Views

System tables

Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No ▾

Test Connection
Save

Filtering the Windows database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.

The automatic installation already comes with the MSSQL Server extension previously configured, only needing to connect with the database. If you are using your own previously configured environment, you will need to enable the MSSQL Server extension in PHP.

If you are using a Microsoft Azure server, see more configuration details [click here](#).

PDO DBLIB

Configurando DBLIB Server No Linux

The automatic installation already comes with the PDO DBLIB previously configured, only needing to install the FreeTDS and unixODBC so that you can do the connection.

Be certified that you already have the PHP extension for PDO DBLIB enabled.

You can install the extension with this command on terminal: `sudo apt-get install php7.0-pdo-dblib`

- access the shell(terminal) with root and install the packages below:
 - distribution: Debian
 - `sudo apt-get install unixODBC unixODBC-dev gcc nano wget make`
 - distribution: Ubuntu
 - `sudo apt-get install unixodbc unixodbc-dev gcc nano wget make`
 - distribution: RHEL\CentOS
 - `sudo yum install unixODBC unixODBC-devel gcc nano wget make`
 - distribution: Suse\OpenSuse
 - `sudo zypper install unixODBC unixODBC-devel gcc nano wget make`
- download FreeTDS:
 - `get http://cdn1.netmake.com.br/download/freetds-0.95.95.tar.gz`
- decompress the FreeTDS:
 - `tar -zxvf freetds-0.95.95.tar.gz`
- enter in the FreeTDS folder:
 - `cd freetds-0.95.95`
- run the command below to compile and install the FreeTDS :
 - `sudo ./configure --with-tdsver=7.4 --with-unixodbc=/usr --disable-libiconv --disable-static --disable-threadsafe --enable-msdblib --disable-sspi --with-gnu-ld --enable-sybase-compat && make && make install`
- edit the FreeTDS configuration file:
 - distribution: Debian\Ubuntu
 - `sudo nano /usr/local/etc/freetds.conf`
 - distribution: RHEL\CentOS
 - `sudo nano /usr/local/freetds/freetds.conf`
 - distribution: Suse\OpenSuse
 - `sudo nano /usr/local/freetds/freetds.conf`
- sample settings:

[MSSQLServer] host = 192.168.254.171 instance = SQLEXPRESS port = 1433 tds version = 8.0

You need to use the name "MSSQLServer" (it's the DNS from FreeTDS) in the option "Server/Host (Name or IP)" to do the connection with your SQLServer database.

Creating a Connection with ScriptCase

- access ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- after accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"















Creating a new connection

- after that, you will see a page with all the database connections that you can create. Select a "MSSQL Server" connection

Create a connection

Select your database system and create a connection for your project.

 MySQL	 Oracle	 MSSQL Server	 Postgres	 DB2	 Informix
 MS Access	 SQLite	 SyBase	 InterBase	 Firebird	 ODBC

Selecting a Linux database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection MSSQL Server

Enter your database connection details for MSSQL Server

← Choose another connection

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver:

Server/Host (Name or IP):

Username: Password:

Database Name:

Connecting with Linux database

- **onnection Name** : Defines the name of your new connection.
- **BMS Driver** : Allows you to select the drivers: Native SRV PDO, Native SRV or ODBC.
- **erver/Host (Name or IP)** : Insert the name or IPINSTANCE of the server where the Linux database is located.
- **atabase Name** : Select the database.
- **ername** : Inform the user to access the Linux database.
- **assword** : Inform the password to access the Linux database.
- **est Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:

Create a connection MSSQL Server

Enter your database connection details for MSSQL Server

← Choose another connection

CONNECTION FILTER ADVANCED

Decimal Separator:

Yes:

Persistent Connection:

Use the schema before table name:

Advanced setup for the Linux database

- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **nicode Converter** - Select if there is going to be conversion of characters to Unicode.
- **ersistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **se the schema before table name** - Allows the use of schema before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER ADVANCED

Show

Tables

Views

System tables

Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No ▾

Test Connection
Save

Filtering the Linux database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.

The automatic installation already comes with the MSSQL Server extension previously configured, only needing to connect with the database.

If you are using your own previously configured environment, you will need to enable the MSSQL Server extension in PHP.

If you are using a Microsoft Azure server, see more configuration details [click here](#).

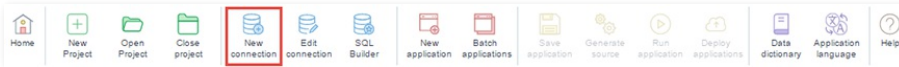
PDO DBLIB

Setting Up PDO DBLIB On Mac OS X

The automatic installation already comes with the PDO DBLIB driver, only needing to connect with the database with ScriptCase.

Creating a Connection with ScriptCase













- access ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- after accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- after that, you will see a page with all the database connections that you can create. Select a "MSSQL Server" connection

Create a connection
Select your database system and create a connection for your project.

 MySQL	 Oracle	 MSSQL Server	 Postgres	 DB2	 Informix
 MS Access	 SQLite	 SyBase	 InterBase	 Firebird	 ODBC

Selecting a Mac OS X database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection MSSQL Server

Enter your database connection details for MSSQL Server

← Choose another connection

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver:

Server/Host (Name or IP):

Username: Password:

Database Name:

Connecting with Mac OS X database

- **Connection Name** : Defines the name of your new connection.
- **DBMS Driver** : Allows you to select the drivers: Native SRV PDO, Native SRV or ODBC.
- **Server/Host (Name or IP)** : Insert the name or IP/INSTANCE of the server where the Mac OS X database is located.
- **Database Name** : Select the database.
- **Username** : Inform the user to access the Mac OS X database.
- **Password** : Inform the password to access the Mac OS X database.
- **Test Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:

Create a connection MSSQL Server

Enter your database connection details for MSSQL Server

← Choose another connection

CONNECTION FILTER ADVANCED

Decimal Separator:

Persistent Connection:

Use the schema before table name:

Advanced setup for the Mac OS X database

- **Decimal Separator** - Select the separator type, between a dot or a comma.
- **Unicode Converter** - Select if there is going to be conversion of characters to Unicode.
- **Persistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **Use the schema before table name** - Allows the use of schema before the name of the tables.
- **Filter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER ADVANCED

Show

Tables
 Views
 System tables
 Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	<input type="text" value="No"/>

Filtering the Mac OS X database

- **How** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **Tables** - Allows to setup the display of Tables from your database.
 - **Views** - Allows to setup the display of Views from your database.
 - **System Tables** - Allows to setup the display of System Tables from your database.
 - **Procedures** - Allows to setup the display of Procedures from your database.

- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.



The automatic installation already comes with the SQLite extension only needing to connect with the database.

If you are using your own previously configured environment, you will need to enable the SQLite extension in PHP.

SQLite PDO

Setting Up SQLite On Windows

The automatic installation already comes with the SQLite driver only needing to connect with the database with ScriptCase.

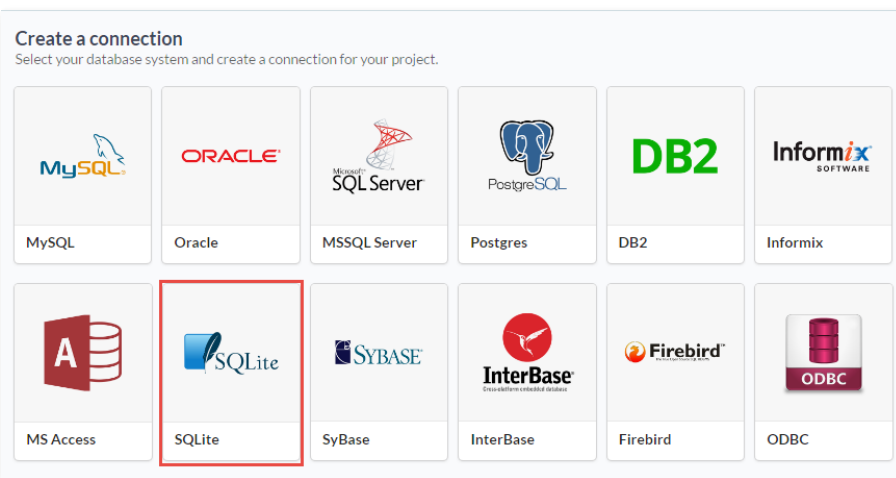
Creating a Connection with ScriptCase

- ccess ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- fter accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- fter that, you will see a page with all the database connections that you can create. Select a "SQLite" connection



Selecting a Windows database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection SQLite

Enter your database connection details for SQLite ← Choose another connection

CONNECTION FILTER ADVANCED

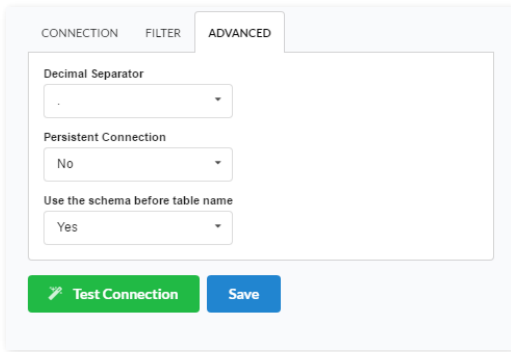
Connection Name: DBMS Driver:

PATH:

Password:

Connecting with Windows database

- **onnection Name** : Defines the name of your new connection.
- **BMS Driver** : Allows you to select the drivers: SQLite PDO.
- **ATH** : Inform the complete path to the SQLite file.
- **est Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:



CONNECTION FILTER **ADVANCED**

Decimal Separator
.

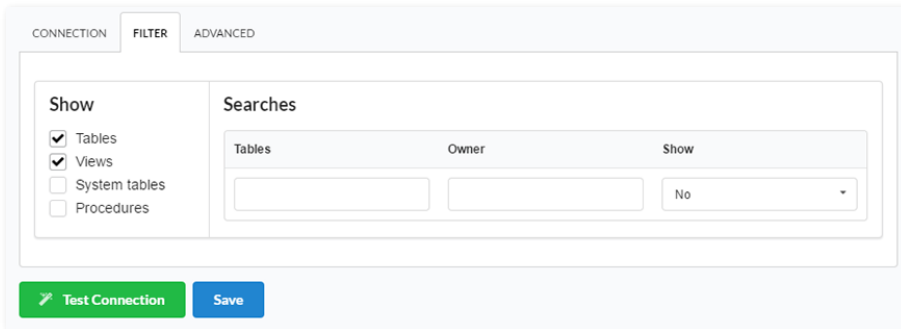
Persistent Connection
No

Use the schema before table name
Yes

Test Connection Save

Advanced setup for the Windows database

- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **se the schema before table name** - Allows the use of schema before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:



CONNECTION **FILTER** ADVANCED

Show

Tables
 Views
 System tables
 Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No

Test Connection Save

Filtering the Windows database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.

The automatic installation already comes with the SQLite extension only needing to connect with the database.

If you are using your own previously configured environment, you will need to enable the SQLite extension in PHP.

SQLite PDO

Setting Up SQLite On Linux

The automatic installation already comes with the SQLite driver, only needing to connect with the database with ScriptCase.

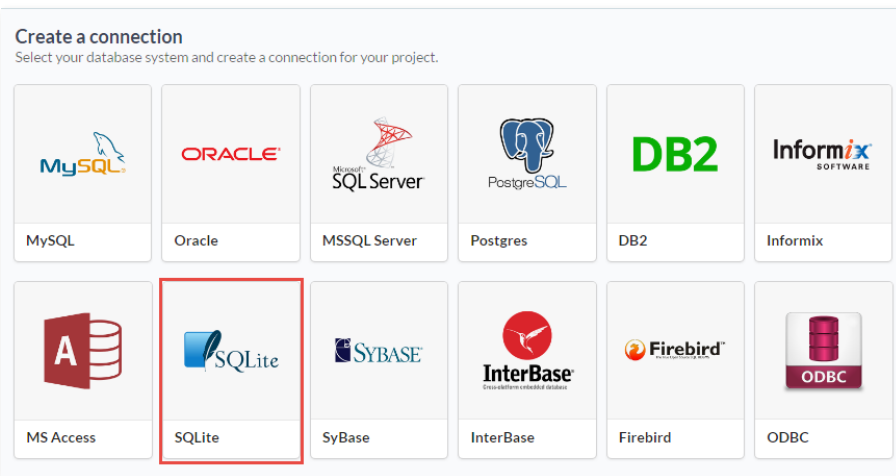
Creating a Connection with ScriptCase

- ccess ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- fter accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- fter that, you will see a page with all the database connections that you can create. Select a "SQLite" connection



Selecting a Linux database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection SQLite

Enter your database connection details for SQLite

[← Choose another connection](#)

CONNECTION FILTER ADVANCED

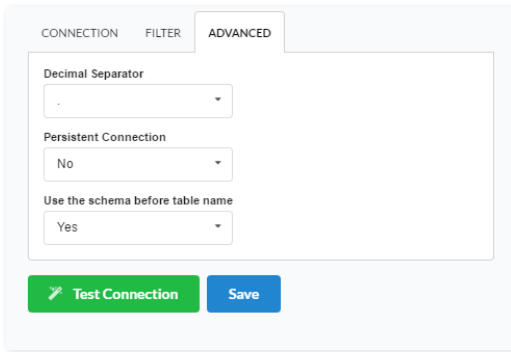
Connection Name: DBMS Driver:

PATH:

Password:

Connecting with Linux database

- **onnection Name** : Defines the name of your new connection.
- **BMS Driver** : Allows you to select the drivers: SQLite PDO.
- **ATH** : Inform the complete path to the SQLite file.
- **est Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:



CONNECTION FILTER **ADVANCED**

Decimal Separator
.

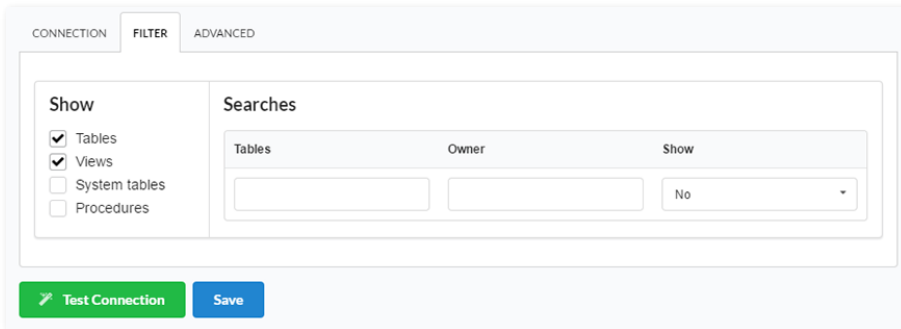
Persistent Connection
No

Use the schema before table name
Yes

Test Connection Save

Advanced setup for the Linux database

- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **se the schema before table name** - Allows the use of schema before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:



CONNECTION **FILTER** ADVANCED

Show

Tables
 Views
 System tables
 Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No

Test Connection Save

Filtering the Linux database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.

The automatic installation already comes with the SQLite extension only needing to connect with the database.

If you are using your own previously configured environment, you will need to enable the SQLite extension in PHP.

SQLite PDO

Setting Up SQLite On MAC

The automatic installation already comes with the SQLite driver, only needing to connect with the database with ScriptCase.

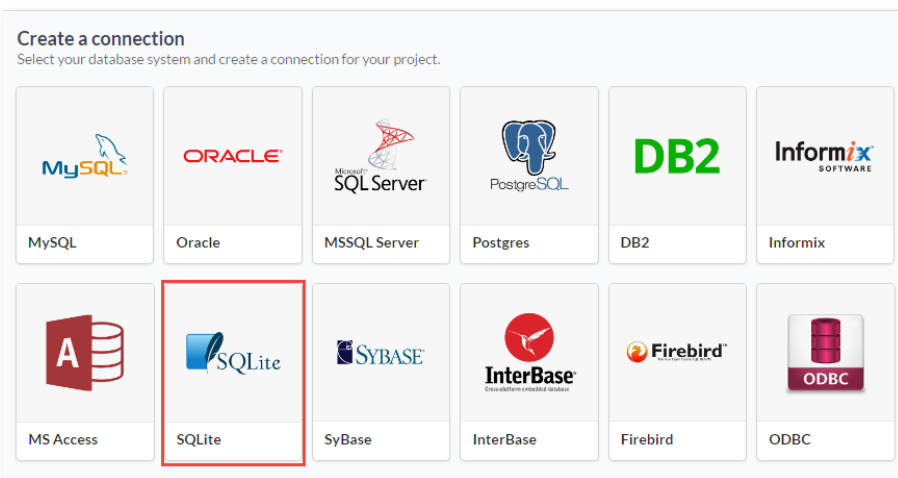
Creating a Connection with ScriptCase

- ccess ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- fter accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- fter that, you will see a page with all the database connections that you can create. Select a "SQLite" connection



Selecting a Mac OS X database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection SQLite

Enter your database connection details for SQLite

[← Choose another connection](#)

CONNECTION FILTER ADVANCED

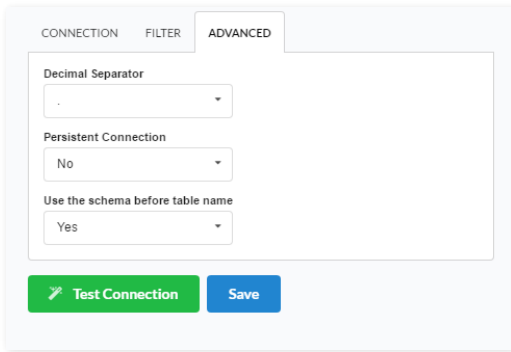
Connection Name: DBMS Driver:

PATH:

Password:

Connecting with Mac OS X database

- **onnection Name** : Defines the name of your new connection.
- **BMS Driver** : Allows you to select the drivers: SQLite PDO.
- **ATH** : Inform the complete path to the SQLite file.
- **est Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:



CONNECTION FILTER **ADVANCED**

Decimal Separator
.

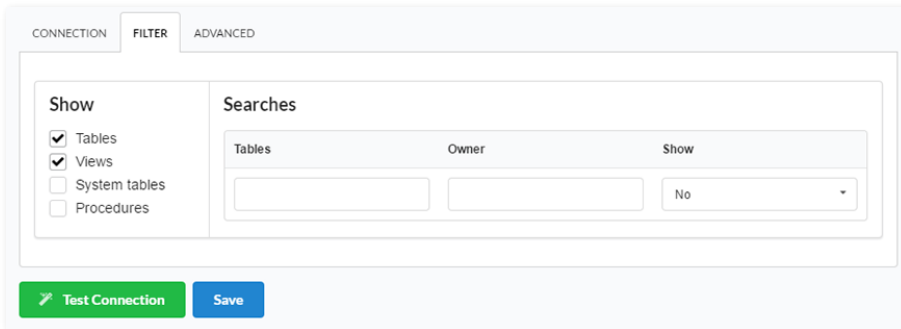
Persistent Connection
No

Use the schema before table name
Yes

Test Connection Save

Advanced setup for the Mac OS X database

- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **se the schema before table name** - Allows the use of schema before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:



CONNECTION **FILTER** ADVANCED

Show

Tables
 Views
 System tables
 Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No

Test Connection Save

Filtering the Mac OS X database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.



The ODBC Generic connection is normally used to create connections with the databases: dBase (.dbf), Dataflex, Paradox, Visual FoxPro, etc.

You only need have the corresponding ODBC driver with the database installed.

If you are using your own previously configured environment, you will need to enable the ODBC extension in PHP.

Generic ODBC

Setting Up ODBC On Windows

The automatic installation already comes with the ODBC only needing to connect with the database with ScriptCase.

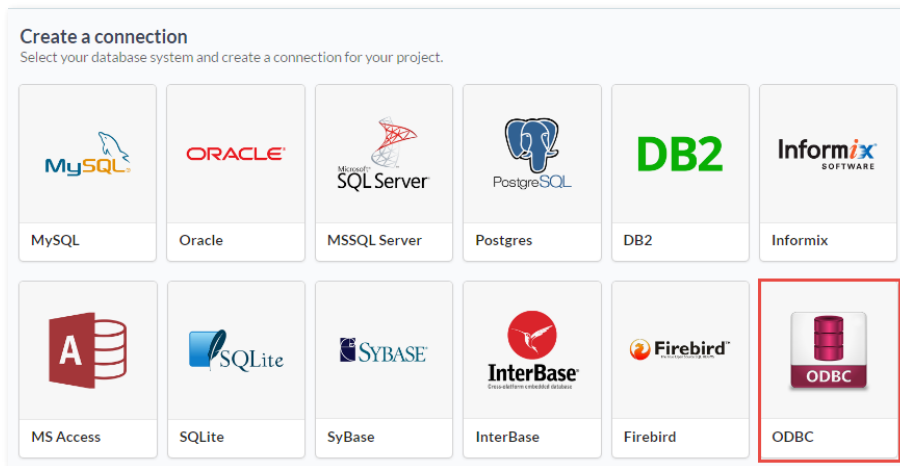
Creating a Connection with ScriptCase

- ccess ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- fter accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- fter that, you will see a page with all the database connections that you can create. Select a "ODBC" connection



Selecting a Windows database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection ODBC

Enter your database connection details for ODBC ← Choose another connection

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver:

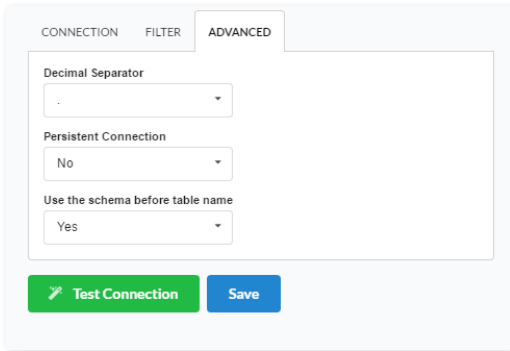
Specific driver: ODBC Name:

Username: Password:

Connecting with Windows database

- **onnection Name** : Defines the name of your new connection.
- **BMS Driver** : Select the Generic ODBC.
- **pecific driver** : Defined as Generic ODBC, allows you to connect to specific Data Sources: DBF, FileMaker, Progress or Visual FoxPro.
- **DBC Name** : On this option, you will inform the name of the Data Source that you have created.
- **ername** : Inform the username of the Windows database, only if there are any.
- **assword** : Inform the password of the Windows database, only if there are any.
- **est Connection** : Displays a message of the status of the connection if successful or not.

- or more options on connecting, click on the **Advanced** tab:



CONNECTION FILTER **ADVANCED**

Decimal Separator
.

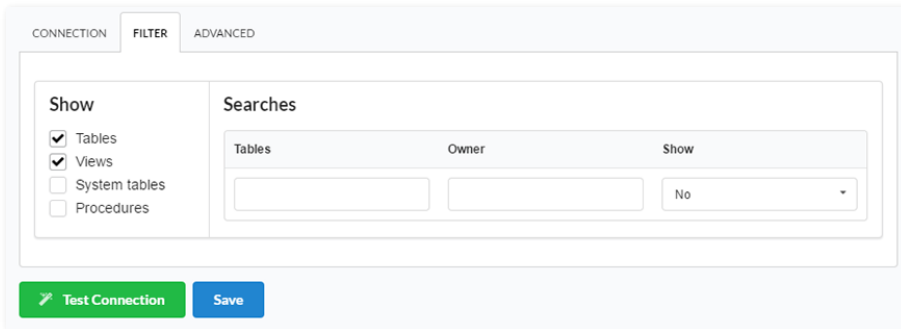
Persistent Connection
No

Use the schema before table name
Yes

Test Connection Save

Advanced setup for the Windows database

- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **se the schema before table name** - Allows the use of schema before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:



CONNECTION **FILTER** ADVANCED

Show

Tables
 Views
 System tables
 Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No

Test Connection Save

Filtering the Windows database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.

The ODBC Generic connection is normally used to create connections with the databases: dBase (.dbf), Dataflex, Paradox, Visual FoxPro, etc.

You only need have the corresponding ODBC driver with the database installed.

If you are using your own previously configured environment, you will need to enable the ODBC extension in PHP.

Generic ODBC

Setting Up ODBC On Linux

The automatic installation already comes with the ODBC only needing to install the UnixODBC driver so that you can do the connection.

UBUNTU\DEBIAN x86_64

- sudo apt-get update

- sudo apt-get install unixodbc-dev unixodbc

CENTOS\RHEL x86_64

- sudo yum update

- sudo yum install unixODBC

UBUNTU\DEBIAN i386 (x86)

- sudo apt-get update

- sudo apt-get install unixodbc-dev unixodbc

CENTOS\RHEL i686 (x86)

- sudo yum update

- sudo yum install unixODBC

Restart the Apache Server:

```
sudo /etc/init.d/apachesc9 restart
```

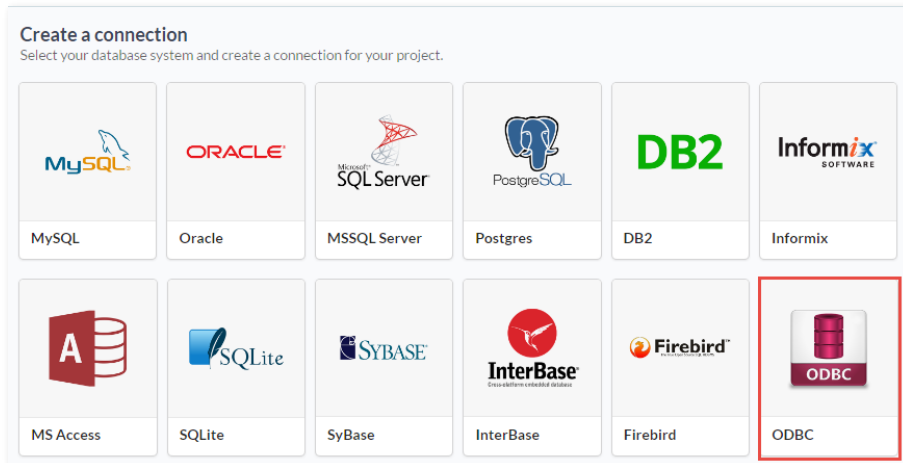
Creating a Connection with ScriptCase

- ccess ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- fter accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- fter that, you will see a page with all the database connections that you can create. Select a "ODBC" connection



Selecting a Linux database connection

- **ew connection** : You will place the information of the connection to your database here:

Create a connection ODBC
Enter your database connection details for ODBC

[← Choose another connection](#)

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver:

Specific driver: ODBC Name:

Username: Password:

[Test Connection](#) [Save](#)

Connecting with Linux database

- **onnection Name** : Defines the name of your new connection.
- **BMS Driver** : Select the Generic ODBC.
- **pecific driver** : Defined as Generic ODBC, allows you to connect to specific Data Sources: DBF, FileMaker, Progress or Visual FoxPro.
- **DBC Name** : On this option, you will inform the name of the Data Source that you have created.
- **ername** : Inform the username of the Linux database, only if there are any.
- **assword** : Inform the password of the Linux database, only if there are any.
- **est Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:

CONNECTION FILTER **ADVANCED**

Decimal Separator:

Persistent Connection:

Use the schema before table name:

[Test Connection](#) [Save](#)

Advanced setup for the Linux database

- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **se the schema before table name** - Allows the use of schema before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER **ADVANCED**

Show

Tables
 Views
 System tables
 Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	<input type="text" value="No"/>

[Test Connection](#) [Save](#)

Filtering the Linux database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.

The ODBC Generic connection is normally used to create connections with the databases: dBase (.dbf), Dataflex, Paradox, Visual FoxPro, etc.

You only need have the corresponding ODBC driver with the database installed.

If you are using your own previously configured environment, you will need to enable the ODBC extension in PHP.

Generic ODBC

Setting Up ODBC On MAC

The automatic installation already comes with the ODBC driver, only needing to create the data source to connect with the database with ScriptCase.

The path of `odbc.ini` is: `/Applications/Scriptcase/components/apache/bin/unixODBC/etc/`

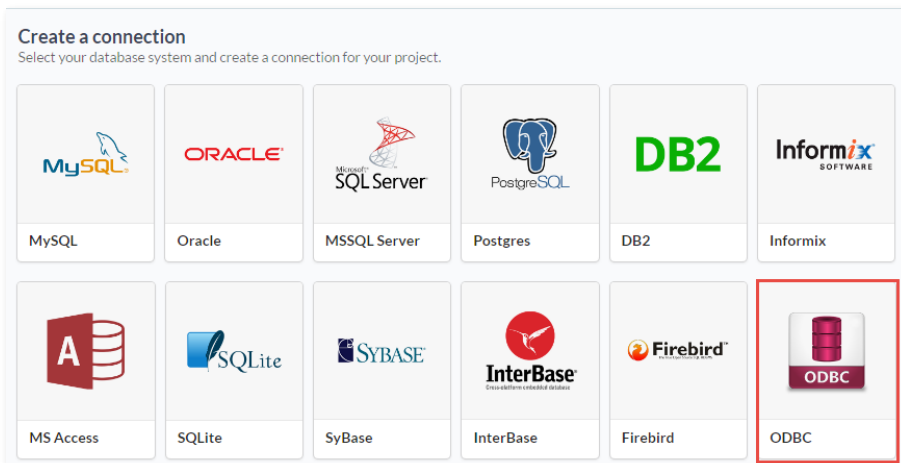
Creating a Connection with ScriptCase

- Access ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- After accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- After that, you will see a page with all the database connections that you can create. Select a "ODBC" connection



Selecting a Mac OS X database connection

- **New connection** : You will place the information of the connection to your database here:

Create a connection ODBC

Enter your database connection details for ODBC ← Choose another connection

CONNECTION FILTER ADVANCED

Connection Name: DBMS Driver:

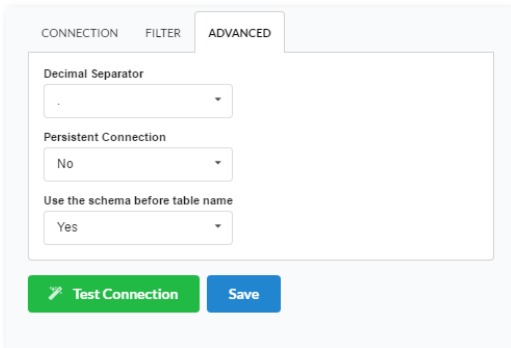
Specific driver: ODBC Name:

Username: Password:

Connecting with Mac OS X database

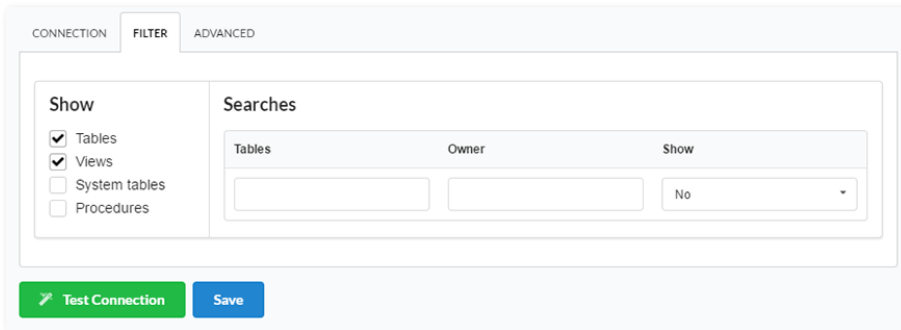
- **Connection Name** : Defines the name of your new connection.
- **DBMS Driver** : Select the Generic ODBC.
- **Specific driver** : Defined as Generic ODBC, allows you to connect to specific Data Sources: DBF, FileMaker, Progress or Visual FoxPro.

- **DBC Name** : On this option, you will inform the name of the Data Source that you have created.
- **sername** : Inform the username of the Mac OS X database, only if there are any.
- **assword** : Inform the password of the Mac OS X database, only if there are any.
- **est Connection** : Displays a message of the status of the connection if successful or not.
- or more options on connecting, click on the **Advanced** tab:



Advanced setup for the Mac OS X database

- **ecimal Separator** - Select the separator type, between a dot or a comma.
- **ersistent Connection** - Persistent Connection are connections that do not close when finishing a script.
- **se the schema before table name** - Allows the use of schema before the name of the tables.
- **ilter** : Accessing this tab, we can setup which tables will be listed on this connection:



Filtering the Mac OS X database

- **how** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **ables** - Allows to setup the display of Tables from your database.
 - **iews** - Allows to setup the display of Views from your database.
 - **ystem Tables** - Allows to setup the display of System Tables from your database.
 - **rocedures** - Allows to setup the display of Procedures from your database.
- **earches** - Allows to define which tables and Owner will be displayed.
 - **ables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **wner** - Allows to define the owner of the tables for listing.
 - **how** - Allows to define what will be displayed or not from the table an owner's setup.

Before proceeding with this tutorial check the architecture of your PHP within the [phpinfo \(\)](#). Accessing the phpinfo of your scriptcase, for example <http://127.0.0.1:8090/scriptcase/info.php>, you will find the PHP architecture.

PHP Version 7.0.15	
System	Windows NT DESKTOP-EB1KEI
Build Date	Jan 17 2017 13:40:02
Compiler	MSVC14 (Visual C++ 2015)
Architecture	x64
Configure Command	cscript /nologo configure.js "--enable-snapshot-build=x64" --enable-openssl --enable-zlib --enable-xml --enable-sockets --enable-mcrypt=static --without-ari...
Server API	CGI/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	C:\Windows

- Architecture **x86** = 32 bits
- Architecture **x64** = 64 bits

If you are using Scriptcase automatic installer the PHP architecture will be the same as the installer you downloaded.

Progress OpenEdge ODBC

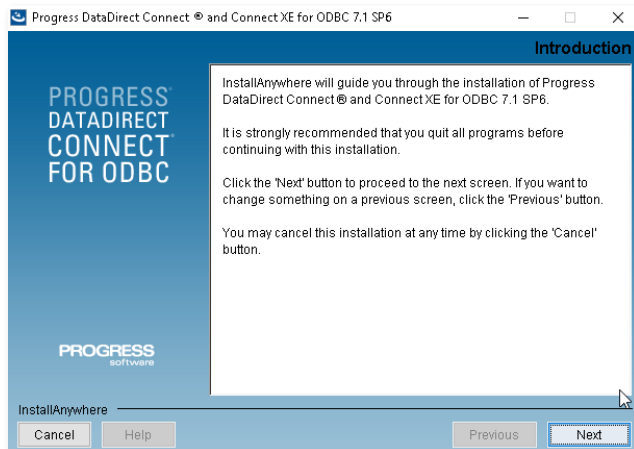
Setting Up Progress Connection Using ODBC On Windows

Requirements

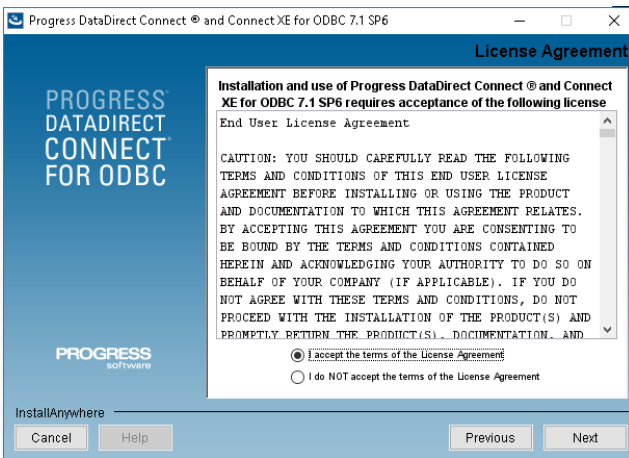
- The Progress Database needs to be installed.
- Have one or more tables created.
- Driver Progress® OpenEdge® ODBC Connector for Windows installed.

Enabling The Extension

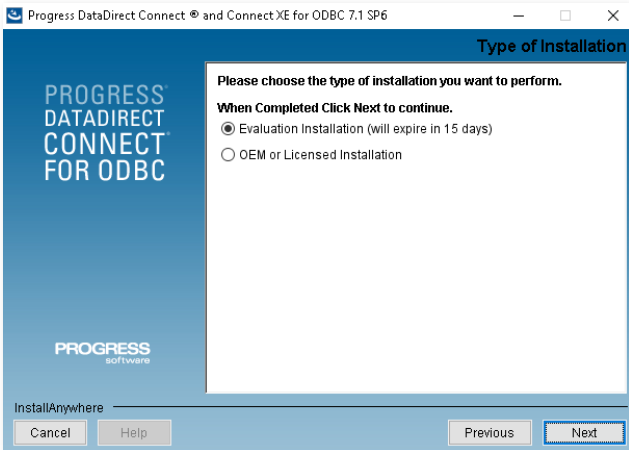
- First we must download the Progress® OpenEdge® ODBC Connector for Windows according to the architecture of your PHP.
 - To download the 64-bit engine [click here](#)
 - To download the 32-bit engine [click here](#)
- At the end of the download, follow the Progress OpenEdge ODBC Connector installation wizard.



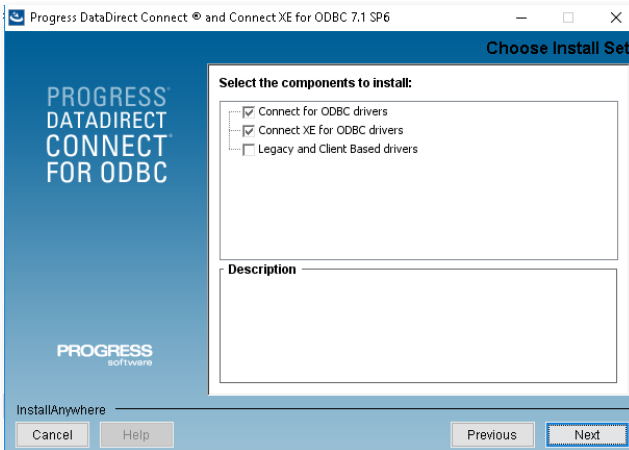
Installing the Progress OpenEdge ODBC Connector.



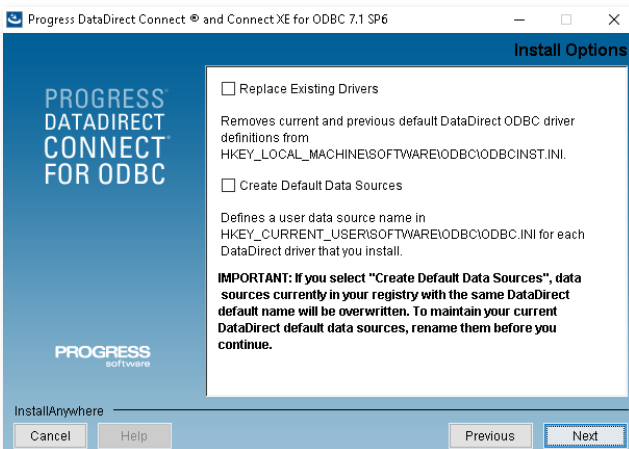
Installing the Progress OpenEdge ODBC Connector.



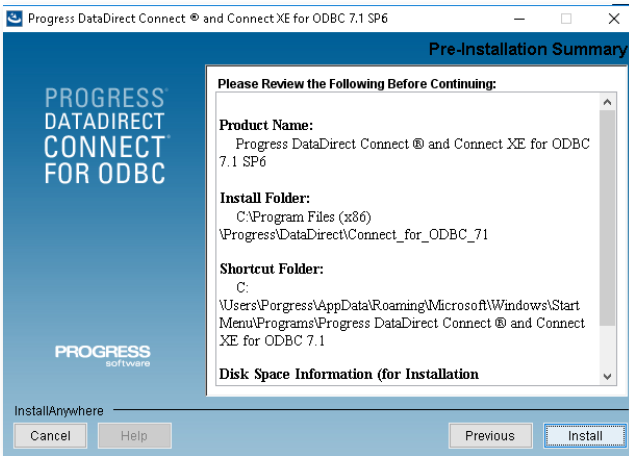
Installing the Progress OpenEdge ODBC Connector.



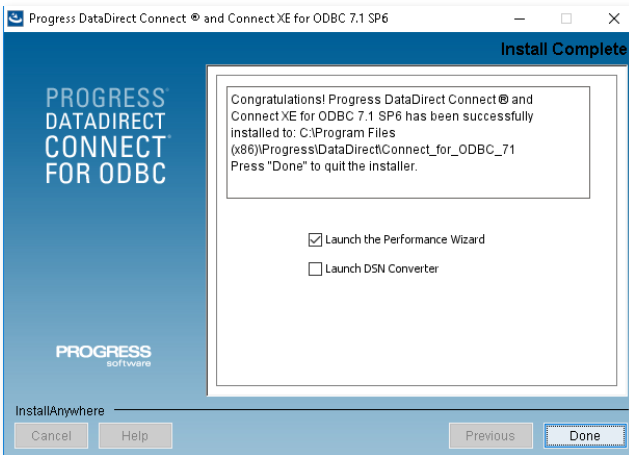
Installing the Progress OpenEdge ODBC Connector.



Installing the Progress OpenEdge ODBC Connector.



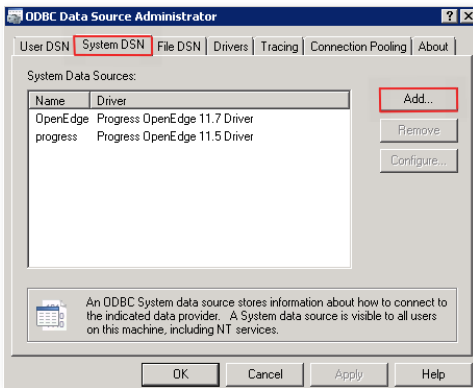
Installing the Progress OpenEdge ODBC Connector.



Installing the Progress OpenEdge ODBC Connector.

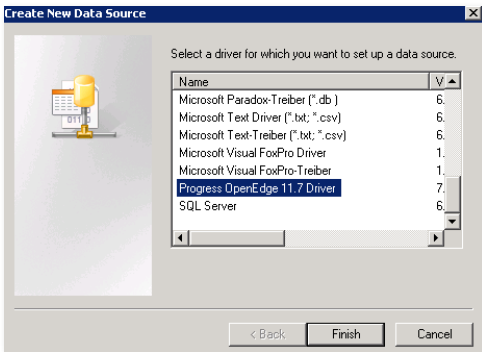
Creating A Data Source (ODBC)

- Go to Control Panel > Systems and Security > Administrative Tools > ODBC Data Sources (32 or 64 bits), according to your PHP architecture.
 - If you have a data source configured, proceed to [Creating a Connection in ScriptCase](#).
- Select the System DSN tab and click Add.



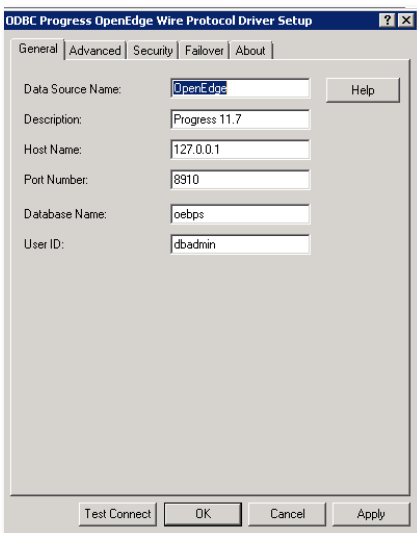
Accessing and Configuring the ODBC Data Source.

- Chose the following driver:
 - Progress OpenEdge 11.7 Driver



Accessing and Configuring the ODBC Data Source.

- After selecting the driver, the following screen will be displayed:
 - Six fields are required for data source configuration:
 - **Data Source Name** : Name of the data source that will be used when creating the connection in Scriptcase;
 - **Description**: A description for your driver;
 - **Host Name**: IP from where the database is installed;
 - **Port Number**: Port Number;
 - **Data Base Name**: Database name that will be used in the connection;
 - **User ID**: User name;



Creating the Data Source.

- After selecting the database, click OK to complete the configuration.
- Returning to the ODBC Data Source Administrator screen you can view the already configured connections.

Creating a Connection with ScriptCase

- Access ScriptCase using the URL <http://127.0.0.1:8090/scriptcase>
- After accessing ScriptCase, access or create a new project, click on the new connection icon or access "Database > New Connection"



Creating a new connection

- After that, you will see a page with all the database connections that you can create. Select a "" connection

Selecting a Windows database connection

- **New connection** : You will place the information of the connection to your database here:

Connecting with Windows database

- For more options on connecting, click on the **Advanced** tab:

Advanced setup for the Windows database

- **Filter** : Accessing this tab, we can setup which tables will be listed on this connection:

CONNECTION FILTER ADVANCED

Show

Tables

Views

System tables

Procedures

Searches

Tables	Owner	Show
<input type="text"/>	<input type="text"/>	No ▾

Test Connection
Save

Filtering the Windows database

- **Show** - Allows the display of the the filters on the tables, views, System Tables and procedures.
 - **Tables** - Allows to setup the display of Tables from your database.
 - **Views** - Allows to setup the display of Views from your database.
 - **System Tables** - Allows to setup the display of System Tables from your database.
 - **Procedures** - Allows to setup the display of Procedures from your database.
- **Searches** - Allows to define which tables and Owner will be displayed.
 - **Tables** - Allows to define a prefix (prefix%) or name of the tables for display.
 - **Owner** - Allows to define the owner of the tables for listing.
 - **Show** - Allows to define what will be displayed or not from the table an owner's setup.

Scriptcase Applications

The Scriptcase offers a wide range of applications that allows the developer to create a complete system. The Scriptcase also allows the integration of the applications with external libraries, boosting its power in the systems development.

See below the available types of applications.

Grid

Application used to display data, this application serves as a report where we can do exports to PDF, XLS, XML, and other formats.

Procedure

Similar to the grid application, this application is available for few databases (MSSQL Server, Oracle and Db2) when selecting the connection, in case the user have a procedure that returns information it will be displayed in this special application.

Form

Application used to insert and update data, there are four different types of orientations.

Tabs

This application allows the creation of tabs where we can put up other applications like grids and forms.

Control

This application is used to create controls in a system, it is not linked to a table, and it needs the creation of one or more fields to its operation. A example of using is in the creation of a login screen.

Menu

It is used to create a hierarchical structure to navigate through the system's applications. There are two types of menu available in Scriptcase, Horizontal Menu and Tree Menu, that differs only in the preview of its items.

Search

Application used to create searches exclusively to forms.

Report PDF

Application used to generate reports using pre set formats, where we can position the data in the report.

Dashboard

Allows the developer to display applications inserted in widgets.

Blank

This application allows the insert of PHP code / HTML and displays the result of its processing. The advantage of using this application is the possibility of using macros and the native integrations with the other applications.

Calendar

Application where we can maintain an event schedule that can be synchronized to the google calendar.


Batch Applications Creation

With this tool, it is possible to create multiple applications (Form and Grid).

When creating grids and forms using the same table, the applications are created with an application link between them, allowing to edit the record from the Grid application.

To start the process of creating the application, you need to select a connection so that the tables can be listed.

Applications Express creation

Connection 

Next, you need to select the tables that are going to be used to create the applications. When selecting the tables, you can define which applications (Forms and Grids) are going to be created.

Tables (3 / 39)

- employees
- employeestate
- employeeterritories
- events
- files_table
- order_details Form Grid
- orders Form Grid
- products Form Grid
- project
- region
- releases

For last, you need to define the name, description and type, in case for the Forms.

Applications Description

Table: order_details (Form)	Name	form_order_details_1	Description	<input type="text"/>
Table: order_details (Grid)	Name	grid_order_details	Description	<input type="text"/>
Table: orders (Form)	Name	form_orders_1	Description	<input type="text"/>
Table: orders (Grid)	Name	grid_orders_1	Description	<input type="text"/>
Table: products (Form)	Name	form_products	Description	<input type="text"/>
Table: products (Grid)	Name	grid_products	Description	<input type="text"/>

Generate Source To edit

- **ame** - Name of the application that is going to be created.
- **escription** - Application Description.
- **ype** - This option is only available for Forms, defines the type of form that is going to be created (Single Record, Multiple Rows, Editable Grid and Editable Grid "View").
- **enerate Source** - Selecting this option all the applications to be created will have their source code generated.
- **o edit** - Selecting this option all the applications created will be open for modifications right after creation.

Restore Applications

This feature allows to restore the project's applications that has been modified recently. When saving the application, ScriptCase stores automatically a limited number (defined in Settings > System Settings on the option Number of automatic application copies) of copies of the application, so that they can be restored. First, we need to select the application that you want to restore.

Choose the application

grid_new ▼

Next » ?

Previously, you need to select the restore point that you desire and also define a name for the restored application.

Choose the restoration point and the application name

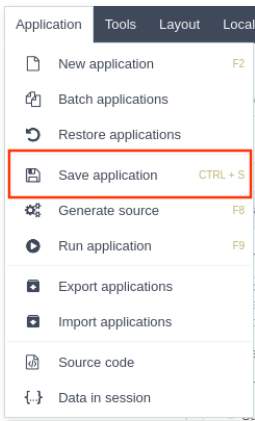
10:23:59 08/08/2017 ▼

grid_new_restore

Next »

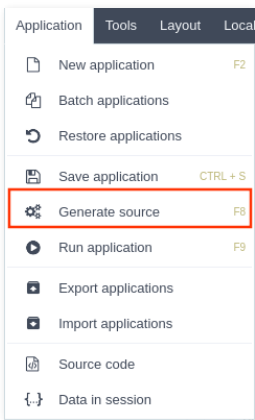
Save Applications

This option saves the application opened in the edition mode. This option can be found inside the applications menu or in the default toolbar, also being possible to use the shortcut CTRL+S.



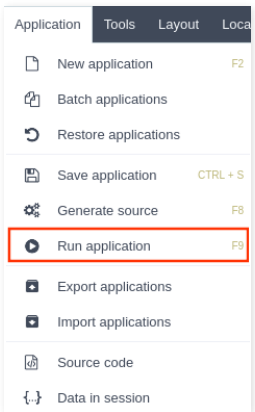
Generate Source

Different from the generate source code found in Project, this option saves and only generate the source code of the opened application in edition mode. This option can be found inside the menu Applications or in the default toolbar, and it is also possible to use the shortcut F8.



Execute Application

Saves and execute the application that is opened in the edition mode. This option can be found inside the Application menu of in the default toolbar, and it is also possible use the shortcut F9.



Export Applications

Every application of a project or some applications can be exported. In this export, not only the application but also the needed archives to its operation of the selected applications are exported.

This option can be found inside the menu **Application**.

Firstly you need to select if you want to export all applications or if you want to select some applications.

Select Applications

All

Select Applications

When select all applications, it will be displayed with the export log and the link to download the generated file.

In this example, we choose to "Select Applications", in this case the applications are listed according to the type or folder.

Select Applications

Applications ?

Visualization Per Type Per Folder

Tab

tab01_1 tab02 tab03

tabs_v9

Blank

sample_blank_application

Calendar

calendar01 calendar02 calendar03

calendar_color_event

Chart

chart05 chart_54_1 chart_54_2

chart_54_3 chart_54_4 chart_area

chart_area_sales chart_bars_2D chart_bars_3D

chart_bubble chart_columns_2D chart_columns_3D

chart_donut chart_employee_by_all chart_gauge

chart_gauge81 chart_line chart_line_step

chart_multi_series chart_orders_by_employee chart_pie_2d

chart_pie_3d chart_pizza chart_pyramid

chart_pyramid81 chart_scatter chart_spline

chart_stacked_bar_3D chart_top_region_orders chart_work_employees

combination_chart sample_chart sample_chart01

sample_chart02 sample_chart03 sample_chart04

tab03_2

Grid

chart01 chart02 chart03

To proceed, select the desired applications and click in export. After this it will be displayed a log of the exported files and a link to download the generated file.

Backup Routine

Report

Time: 0:00:06,00

Items: 4337

MODULE	ITEMS	TIME
Buttons	19	0:00:00,04
Icons	4	0:00:00,01
Images	31	0:00:00,08
Theme	1	0:00:00,00
Footer Templates	2	0:00:00,00
Body Templates	33	0:00:00,05
Header Templates	2	0:00:00,01
PHP Libraries	1	0:00:00,01
Languages	50	0:00:00,13
Projects	1	0:00:00,00
Version	1	0:00:00,01
Connections	1	0:00:00,00
User	1	0:00:00,00
Application	337	0:00:01,88
Fields	3615	0:00:01,53
Events	237	0:00:00,60
Zip File	1	0:00:01,62

Backup file

sc9_201708081048_export_samples.zip

Import Applications

Allows the developer to import applications created in the same version of Scriptcase. This option can be found inside the menu **Application**.

After the import application screen is displayed, select the file you want to import.

The options are:

Overwrite - Overwrites the project's files by the files that are being imported. **Don't overwrite** - Keeps the project's files, ignoring the files that are being imported. **Rename** - Keep the project's and import the files selected with the defined names.

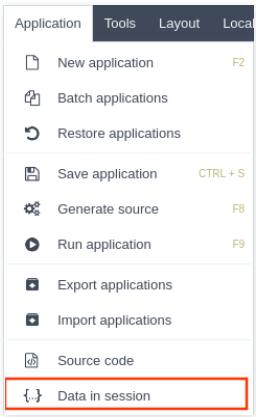
In the next step, show a summary that will be restored.

Source code

In this option is possible to see the application's source code after it generates.

Session data

In this option, it is possible to see every session variable available in the development environment, it is the session tree that Scriptcase stores.



Initial Module

Allows you to set in which mode the application will start when executed.

INITIAL MODULE

Select the Grid module that will be displayed when running the application:

- Search
- Grid
- Summary
- Chart
- Print
- PDF
- Word
- Excel
- RTF
- XML
- CSV


Grid Application Modules


- **ilter** : Application will be started by the Filter view, so you can filter the records before the next application, that can be a Grid, PDF, Summary... you can configure these options within the Filter's settings.
- **rid** : Application will be started by the Grid itself, this is the default option.
- **ummary** : Application will be started by the Summary. To use the Summary as initial application it's mandatory to create at least one Group, using the Static Group By settings.
- **hart** : Application will be started by the Chart. To use the Chart as initial application it's mandatory to create at least one Group, using the Static Group By settings.
- **rint** : Application will be started by the print mode, according to the select command.
- **DF** : Application will be started with the option to view or download a PDF file.
- **ord** : Application will be started with the option to view or download a Word file(.doc or .docx).
- **xcel** : Application will be started with the option to view or download an Excel file (.xls or .xls).
- **TF** : Application will be started with the option to view or download a RTF file (.rtf).
- **ML** : Application will be started with the option to view or download a XML file.
- **SV** : Application will be started with the option to view or download a CSV file.


Filter


APPLICATION MODULES


Select the modules that will be part of the Grid and set its parameters:


 Search


 Grid


 Detail


 Summary


 Chart

Filter module	Grid ▼	Application module where search is applied.
Use Iframe	No ▼	Use Iframes to display the Search and the Grid on the same page.
Show Results	Yes ▼	Display the search results on the same page when loading the application for the first time using Iframes.
Iframe Height	1500	Iframe height in pixels where the grid will be displayed.
Search Modal	No ▼	Display the search in a modal window.
Search Modal Height	0	Height search modal.
Search Modal Width	0	Search modal width.
Table Width	0	Width value for the application table.
Table Width Unit	Automatic ▼	Measure unit for the width.

Advanced Settings

Grid Filter Settings


- **ilter Module** : This option configures which application will be called after the Filter.
- **se Iframe** : This option configures the filter to be displayed within an iframe (displaying the Filter itself and the search results on the same page). This option is available only when the initial module is set to be the Filter.
- **ide Grid Header** : Hide the Grid's header when using the Search in the iframe.
- **how Results** : Display the search results on the same page when loading the application for the first time when the "Use Iframe" option is set as "Yes", otherwise, the results will be displayed only after the search.
- **frame Height** : Iframe height, in pixels, used to display the search results.


- **earch Modal** : This option configures the Filter Application to open in a modal window.
- **earch Modal Height** : Search Modal box height (in pixels). Option available only when the Search Modal box is enabled.
- **earch Modal Width** : Search Modal box Width (in pixels). Option available only when the Search Modal box is enabled
- **able Width** : Filter application table width. This value can be in percent, pixel or automatic (set by Scriptcase). If you change the value you will also need to set the "Table Width Unit"
- **able Width Unit** : This option configures the unit for the table width (percent, pixel or automatic). The value itself must to be set within the option "Table Width".


Grid


APPLICATION MODULES


Select the modules that will be part of the Grid and set its parameters:


 Search


 Grid


 Detail


 Summary


 Chart

Orientation	<input type="text" value="Horizontal"/>	Grid records' orientation (Horizontal, Vertical, Slide or User Defined).
Fixed label	<input type="text" value="Yes"/>	This option will freeze the column labels on top of the screen during the page scrolling.
Pagination	<input type="text" value="Partial"/>	Pagination method for the grid records.
Lines Per Page	<input type="text" value="10"/>	Number of record lines per page.
Infinite Scroll Increment	<input type="text" value="5"/>	Number of rows to load when loading new records
Open windows using modal	<input type="text" value="Yes"/>	For Grids apps with the Infinite Scroll pagination enabled, this option will set the links between applications and details of the Grid, that are configured to open in iframe, to open using a modal instead.
Maintain records	<input type="text" value="Yes"/>	When you navigate to another window and go back, it will keep the amount records already displayed
Grid height	<input type="text" value=""/>	It defines the height of the Grid in pixels, for when "Infinite Scroll" option is activated. The infinite scrolling will occur for the whole page if this height is not configured.
Table Width	<input type="text" value="100"/>	Set the application width.
Table Width Unit	<input type="text" value="Percent"/>	Measure unit used for the application width.

Advanced Settings


Grid Settings


- **rientation** : This option configures the Grid records' orientation (Horizontal, Vertical, Slide or User Defined). When using the "User Defined" option you can design the HTML manually inside the option "Layout » HTML templates" and select within the Layout settings.
- **ixed Label** : This option will fix the columns' label at the top of the page (it is only available when the Grid records' orientation is set as Horizontal).
- **agination** : This option sets the Grid paging type: Partial (pagination according to the amount of records set per page), Total (displays all records) or Infinite Scroll (automatic scrolling according to the increment)
- **ines Per Page** : This option sets the amount of records per page for the Grid
- **nfinite Scroll Increment** : This option sets the number of rows displayed on each increment of new records. It is available only when paging is configured with Infinite Scroll.
- **pen windows using modal** : For Grids application with the Infinite Scroll paging enabled, this option will set the links between applications and details of the Grid, that are configured to open in iframe, to open using a modal instead.
- **aintain records** : It configures if the amount of records displayed will be preserved when the application navigates to another window and go back.
- **rid height** : It sets the height of the query in pixels. If it is empty
- **able Width** : Grid application table width. This value can be in percent, pixel or automatic (set by Scriptcase). If you change the value you will also need to set the "Table Width Unit"
- **able Width Unit** : This option configures the unit for the Grid table width (percent, pixel or automatic). The value itself must to be set within the option "Table Width".


Detail


APPLICATION MODULES


Select the modules that will be part of the Grid and set its parameters:


 Search


 Grid


 Detail


 Summary


 Chart

Display Detail	<input type="text" value="In another page"/>	Detail record display mode
Alignment	<input type="text" value="Left"/>	Field alignment for detail page.
Detail Width	<input type="text" value="0"/>	Width value for the application detail
Width of the modal	<input type="text" value=""/>	Width of the modal window in pixels.
Height of the modal	<input type="text" value=""/>	Height of the modal window in pixel
Detail Width Unit	<input type="text" value="Automatic"/>	Measure unit used in the width.

Advanced Settings

Grid Detail settings

- **isplay Detail** : Allows you to set where the detail will open within the Grid Application * **Beside the Grid** : Displays the Grid Details to the right of the record, in the same window where Grid is being displayed. * **Below the Grid** : Displays the Grid Details below the Grid records, in the same window where Grid is being displayed. * **In another page** : Displays the Grid Details in


another page, replacing the Grid view. * **In another window** : Displays the Grid Details in a separated browser window. * **Modal** : Opens a pop-up window to the display of the Grid Detail.


- o **Alignment** : Using this option you can set an alignment (center, right or left) for the Grid Detail when it's using the "Display Detail" setting as "Beside the Grid" or "Below the Grid".
- o **Detail Width** : This option sets the Grid Detail width when it's using the "Display Detail" setting as "Beside the Grid", "Below the Grid", "In another page" or "In another window".
- o **Width of the modal** : Allows you to customize the Grid Detail width when it's using the "Display Detail" option as "Modal".
- o **Height of the modal** : Allows you to customize the Grid Detail height when it's using the "Display Detail" option as "Modal".
- o **Detail Width Unit** : This option configures the unit for the "Detail Width" (percent, pixel or automatic).


Summary


APPLICATION MODULES


Select the modules that will be part of the Grid and set its parameters:


 Search


 Grid


 Detail


 Summary


 Chart

This module will not be generated because the grid has no Group By configured.
Create at least one Group By if you want to use this module.

Summary Display	<input type="text" value="On another page"/>	Summary display page.
-----------------	--	-----------------------

[Advanced Settings](#)


Grid Summary Settings


- **Summary Display** : Sets the Grid Summary display option (On another page, On the last page, On every page). The Summary Application Module is only available when a group is created in the Group By settings.


Chart


APPLICATION MODULES


Select the modules that will be part of the Grid and set its parameters:


 Search


 Grid


 Detail


 Summary


 Chart

This module will not be generated because the grid has no Group By configured.
Create at least one Group By if you want to use this module.

Charts display mode	<input type="text" value="New window"/>	Charts display position.
Display before summary	<input type="text" value="No"/>	Display chart before summary.
Number of columns	<input type="text" value=""/>	Number of charts per line (one chart in each column).
Margin	<input type="text" value="20"/>	Margin between the charts.
Horizontal alignment	<input type="text" value="Left"/>	Charts horizontal alignment.
Vertical alignment	<input type="text" value="Top"/>	Charts vertical alignment.
View settings in the chart window	<input type="text" value="Yes"/>	Allows you to view the settings in same window as the ch:

[Advanced Settings](#)

Grid Chart settings

- **Charts display mode** : Sets the Chart Summary display option (New window, Same page, Other page).The Chart Application Module is only available when a group is created in the Group By settings.
- **Display before summary** : Sets whether the Chart is displayed above or below the Summary. Available when the Charts display mode is configured in the Same page.
- **Number of columns** : Sets the Chart number of columns per line (one in each column). Available when Charts display mode is configured in the Same page.
- **Margin** : Sets the Charts horizontal margin. Available when Charts display mode is configured in the Same page.
- **Horizontal alignment** : Sets the Charts horizontal position (Left, Right, Center). Available when Charts display mode is configured in the Same page.
- **Vertical alignment** : Sets the Charts vertical position (Top, Center, Bottom). Available when Charts display mode is configured in the Same page.
- **View settings in the chart window** : Sets the chart settings display on the same page. Available when the Char display mode is configured on Other page or new window.

RELATED VIDEOS

- [Advanced Reports](#)
- [Initial Modules of a Grid](#)

GRID SETTINGS	
ATTRIBUTE	VALUE
Friendly URL	<input type="text"/>
Display Line Number	<input type="checkbox"/>
Display Titles	<input checked="" type="checkbox"/>
Line break in title	<input type="checkbox"/>
Horizontal Alignment	Center ▾
Margins	<input type="text"/> Up <input type="text"/> Down <input type="text"/> Right <input type="text"/> Left
Alignment	Left ▾
Table Columns	Automatic ▾
Refresh Interval	<input type="text" value="0"/>

Grid application advanced settings

- **riendly URL** : This field allows you to change the URL that will be called by the application. Allowed characters are the same available on URLs: a-z, A-Z, 0-9, -_. This option can also be changed on the home screen, on the "Friendly URL" column at the applications list.
- **isplay Line Number** : This option sets whether to display or not the sequence number of each Grid row.
- **isplay Titles** : This option sets if the column title (column label) will be displayed or not.
- **ine break in title** : This option sets if the column line title will break or not.
- **argins** : Sets the application margins in pixels (up, down, right and left).
- **ignment** : Allows you to set the fields alignment for when the Grid orientation is set to Vertical or Slide.
- **able Columns** : Sets the column widths type: Provides (It will assume the informed width values at field level in the configuration of the visualization), Calculated (Calculate size according to type and the field size) and Automatic (The alignment will be according to the browser criteria).
- **efresh Interval** : Allows you to set a reload interval for the page, in seconds. When is set as zero, there will be no page reload.

This interface is useful for editing the field settings and their position to display.

Fields	Label	Datatype	Line Break	Line Break PDF	Title Horizontal Alignment	Text Alignment	Font Color
PAGE: PAG1							
BLOCK: GRID_ORDERS							
 orderid	Orderid	Integer	<input type="checkbox"/>	<input type="checkbox"/>	Right	Right	<input type="text"/>
 customerid	Customer Name	Text	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Left	Left	<input type="text"/>
 employeeid	Employeeid	Integer	<input type="checkbox"/>	<input type="checkbox"/>	Right	Right	<input type="text"/>
 orderdate	Orderdate	Date	<input type="checkbox"/>	<input type="checkbox"/>	Center	Center	<input type="text"/>
 requireddate	Requireddate	Date	<input type="checkbox"/>	<input type="checkbox"/>	Center	Center	<input type="text"/>
 shippeddate	Shippeddate	Date	<input type="checkbox"/>	<input type="checkbox"/>	Center	Center	<input type="text"/>

Fields

Allows accessing the field settings (pencil icon on the left). You can change the field position by dragging them to the desired position. Drag a field to “fields not displayed” if you don’t want it in the app.

Label

Defines the title of a field in the app. For example: if the field name in the database is fld_txt_customer_name, you can display the label “Customer Name”.

Data Type

It informs the data type of the field.

Line Break

Allows the Line Break in the records when the field text is greater than the column width.

Line Break PDF

Allows the Line Break when the user exports the app as a PDF.

Title Horizontal Alignment

Defines the horizontal alignment of the field label.

Text Alignment

Defines the horizontal alignment of the field text.

Font Color

Configure the text color for the field.

This interface allows to sort the fields positioning.

Select the fields that will be displayed.

Available Fields	Selected Fields (Pag1)
orderid	form_orders
customerid	orderid
employeeid	customerid
orderdate	employeeid
requireddate	orderdate
shippeddate	requireddate
shipvia	
freight	
priceorder	
shipcountry	
shipregion	
shipstate	
shipcity	
shipname	
shipaddress	
shippostalcode	

Save Restore

Field Positioning

The left box has a list of all fields available in the application. Here you must select which fields you want to display in the app.

The right box contains the selected fields displayed in the application.

Use the arrows buttons between the boxes to select or deselect a field.

The application toolbar has two segments: Top and Bottom, in a way that is possible to define to display buttons into both areas. Those areas work independently, allowing them to display the same button, for example.

It's also possible to select the buttons and their position if the application is running on a mobile device.

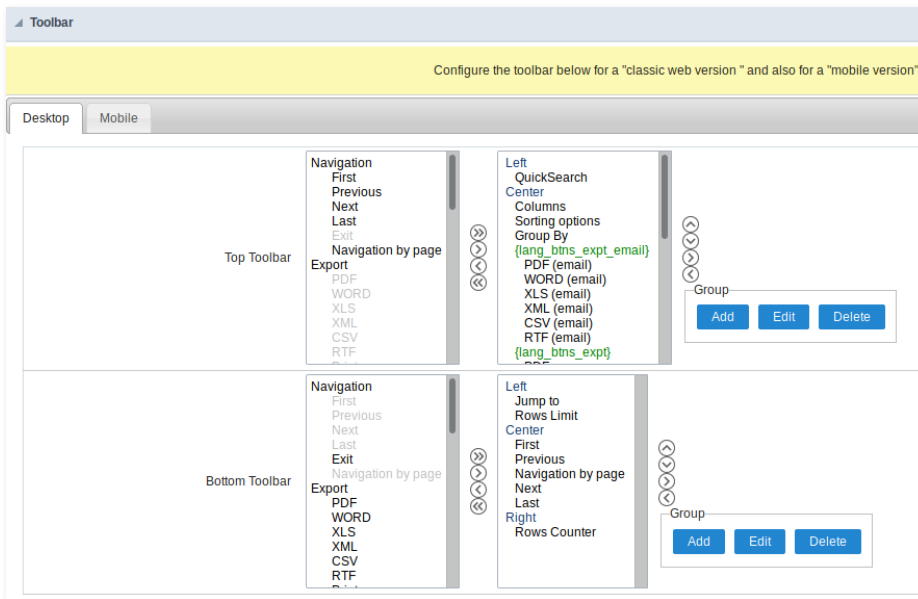
Toolbar

Desktop

Here we must inform the toolbar settings for the "Classic Web Version" mode and which buttons are available in the application when accessed from a **Desktop** environment.

Mobile

Here we must inform the toolbar settings for the "Mobile Version" mode. That is which buttons are available in the application when accessed from a **Mobile** dispositive.



Navigation:

Buttons relative to the navigation of the application.

Next	Move to the next page that can be a single record or a list of records.
Previous	Returns to displays the previous page records or a single record.
First	Move to the First page or record
Last	Move to the Last page or record
Exit	Close the application
Navigation by page	Displays a "page-number" navigation bar. Example: 1 2 3 4 5

Export:

Groups the options relative to the generated exports. Scriptcase generates the following export formats:

PDF	Generates a complete Report with all the data of the application in a PDF format.
WORD	Generates a complete Report with all the data of the application in a WORD format.
XLS	Generates a complete Report with all the data of the application in an EXCEL format.
XML	Generates a complete Report with all the data of the application in an XML format.
CSV	Generates a complete Report with all the data of the application in a CSV format.
RTF	Generates a complete Report with all the data of the application in an RTF format.
Print	Creates an HTML with the records ready for printing.

Export by Email:

Groups the options relative to the emails exports. Scriptcase generates the following export formats:


PDF (Email)	Sends by Email a complete report with all the data of the application in a PDF format.
WORD (Email)	Sends by Email a complete report with all the data of the application in a WORD format.
XLS (Email)	Sends by Email a complete report with all the data of the application in an EXCEL format.
XML (Email)	Sends by Email a complete report with all the data of the application in an XML format.
CSV (Email)	Sends by Email a complete report with all the data of the application in a CSV format.
RTF (Email)	Sends by Email a complete report with all the data of the application in an RTF format.

Others:

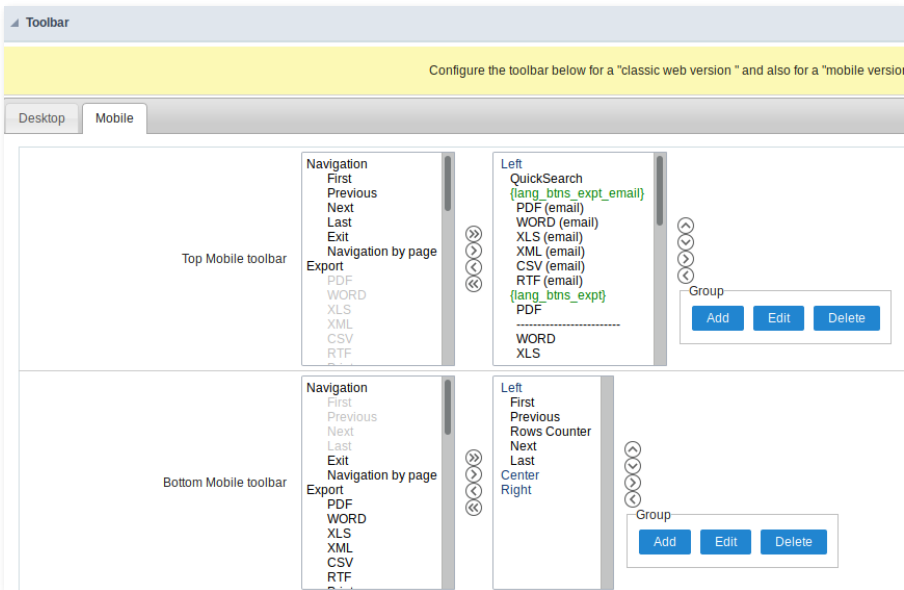
Other options available in the Grid application.

Jump to	Move to the informed page.
Rows Limit	It is a Combobox that defines the number of rows per page.
Search	Goes to the Search Form to filter the records.
Dynamic Search	It displays the fields of the search to filter the records.
Columns	Allows to include or remove columns of the Grid on the fly.
Sorting Options	Allows selecting the order of the records based on the field.
Group By	Allows to select or change a Group By rule on the Grid.
Save Grid	Allows saving the current state of the application. For example, in the advanced search, you can save the search data for further use.
Quick Search	Allows to perform a quick search in the records of the application.
Gantt	It displays a Gantt chart, if it was previously set.
Summary	It displays a summary with the synthetical data of the records.
Form Buttons	It displays the Form Buttons when there's an Application Link from the Grid to a Form.
Languages	It displays a Combobox with the names available, defined in the project properties.
Themes	It displays a Combobox with the themes available, defined in the project properties.
Rows Counter	It displays the number of records retrieved by the application.
HelpCase	Displays a button to open the help page.

Separator:

	Displays a line separating the buttons, when used the Group Buttons.
--	--

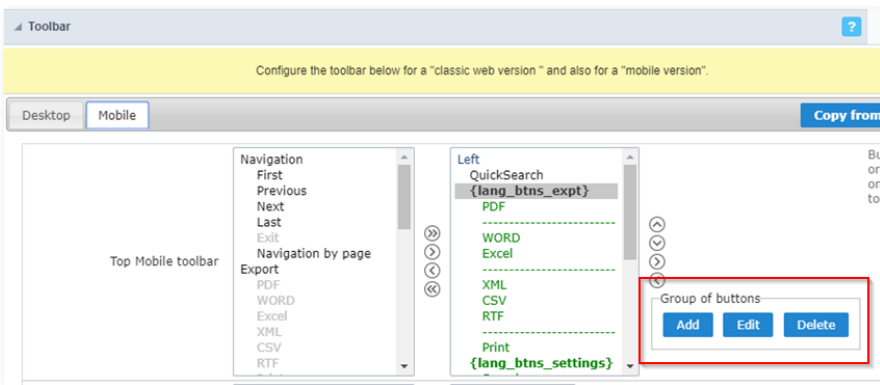
Toolbar Mobile



Contains the same options as the **Desktop** version, adding only the item "Copy from Desktop", on which, when clicked, it copies the items from the **Desktop** toolbar to the **Mobile** toolbar.

Buttons Group

The **Group** option allows you to group a set of buttons of the application toolbar to display them as a dropdown, for example.



Add

Add a new group of buttons.

Edit

Edit an existent group of buttons.

Delete

Delete the selected group of buttons.

When you press the **Add** or **Edit** option, you can see the settings to configure the grouper:

Edit

DISPLAY AS
 Dropdown

DROPDOWN LIST THEME NAME
 Application Theme group_4

LABEL HINT \ TITLE
 Group

IMAGE BUTTON TYPE
 Button

DISPLAY DISPLAY POSITION
 Only Text Text on the right

Cancel Add

Display As

Allows displaying the group button as **Dropdown** or **Side by Side**.

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print Exit

Customer ID	Company Name	Phone	Credit Limit
dos		30074321	\$3,367.41
brilla		55554729	\$7,371.95
		55553932	\$6,757.53

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print Exit

Dropdown List Theme

Allows defining the Dropdown theme selecting between **Application theme** and **Button theme**.

Name

Allows defining a name for the button group.

Label

It is the displayed name for the button group in the application.

Hint\Title

Displays a hint to the end-user when the mouse is on the group of buttons.

Button Type

Allows displaying the button group as a Button, Image, or Link.

Image

Allows selecting an image for the button.

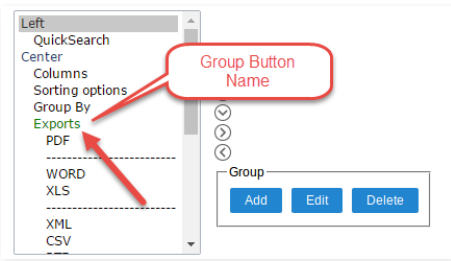
Display

Defines if the button displays only Text, only image, or both.

Display Position

Defines the position of the Text or Image (Text to the right, Image to the right).

After creating a button group, you need to move the grouped buttons below of the Button Group and then move them to the right. Like the image below:



Button Settings

Button	Label	Hint
QuickSearch		
Dynamic Search		
Insert		
Cancel		
Update		
Delete		
Exit		
Jump to		
First		
Previous		
Navigation by page		
Next		
Last		
Rows Counter		
New		

Button:

It displays the buttons available in the application.

Label:

Allows defining the label of the buttons to display for the users.

Hint:

Allows defining the buttons hint to display for the users.

Application Hotkeys

Scriptcase allows creating shortcut keys to your applications. You can select a predefined template or create specific actions for an application.

VALUE	DESCRIPTION
<input checked="" type="checkbox"/> toggleHotkeys();	Define if the application will use hotkeys
SC_DefaultHotkeys <input type="button" value="↻"/> <input type="button" value="✎"/>	Select the hotkey template from previously created schemas
<input type="button" value="Clear"/> <input type="button" value="+"/>	
ACTION	KEYBINDING
No hotkeys configured	
<input type="button" value="Clear"/> <input type="button" value="+"/>	

Use hotkeys

Defines if the application uses hotkeys. When you enable this option, the default shortcut keys settings are disabled.

Hotkeys template

Select the [hotkey template](#) previously created.

Action

Selects the triggered action when pressing the selected key.

Keybinding

Selects the keys responsible for executing the chosen action.

Add "+"

Adds a new action on the keys list.

Clear

It clears the selected hotkeys preference.

Options

ATTRIBUTE	VALUE
Display Summary	<input type="checkbox"/>
Format Row Counter	<input type="text" value="{lang_othr_smry_info}"/>
The number of links displayed	<input type="text" value="5"/>
Jump to	<input type="text" value="Page"/>
Records per page	<input type="text" value="10,20,50"/>
Toolbar buttons	<input type="text" value="Modal"/>

Display Summary:

Allows to display the summary button or not.

Row Counter:

Allows to display the Row Counter or not.

Example: (1 to 10 of 200)

The Number of Links Displayed:

Defines the number of links per page, when the navigation option is disabled.

Jump To:

Move to the informed record page.

Records by page:

Allows defining the number of records to be exhibited for each page. To show all records use the option "all".

Example: 10,20,30, all

Toolbar Buttons:

The type of view for the Toolbar buttons (A DIV below the toolbar, Modal).

PDF Settings

PDF General Settings

General settings	
ATTRIBUTE	VALUE
Configurable PDF	<input checked="" type="checkbox"/>
Grid Export modules	<input checked="" type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Summary export modules	<input type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Export with password	<input type="checkbox"/>
Select columns	<input type="checkbox"/>
Open PDF Directly	<input type="checkbox"/>
PDF Chart depth output selection	<input checked="" type="checkbox"/>
Selecting page break per level	<input type="checkbox"/>

Configurable PDF

It allows the end-user to configure the PDF parameters on the fly.

Export Modules of Grid

This option allows the developer to choose which modules are available in the grid export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
---------------------	---

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Summary Export Modules

This option allows the developer to choose which modules are available in the summary export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid
	<input type="checkbox"/> Summary
	<input checked="" type="checkbox"/> Chart

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Export with password

This option allows the developer to set a password to export the file.

Export with password	<input checked="" type="checkbox"/>
Export with password	<input type="text"/>

Password

It allows the developer to set the requested password when exporting the file.

Runtime Example:

PDF settings	
GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Print settings	Color
Page Layout	A4 (210 X 297 mm)
Orientation	Portrait
Other options	<input checked="" type="checkbox"/> Generate Bookmarks <input checked="" type="checkbox"/> Displays the header on all pages <input checked="" type="checkbox"/> Displays the title on all pages
Password

Select columns

It allows the user has the option to choose the columns he wants to export at runtime.

PDF settings	
Select Columns	
<input type="checkbox"/>	Accountid
<input type="checkbox"/>	Accountdescription
<input type="checkbox"/>	Accountorder

Open PDF Directly

Opens the generated the PDF without the need to display an intermediate page with a link to download the file. **Option available when the option "Create Charts" is disabled.**

PDF Chart depth output selection

It allows the end-user to decide on the fly the depth of the Group By to exports.

Selecting page break per level

It allows the end-user to define the break level to skip to the next page in the generated PDF.

PDF Settings

PDF Settings	
ATTRIBUTE	VALUE
	<input type="checkbox"/>
PDF Orientation	Portrait ▾
PDF Format	A4 (210 X 297 mm) ▾
Print Type	Both ▾
Records per Page	<input type="text" value="0"/>
Rows per Page in Summary.	<input type="text" value="0"/>
Complete Lines Until Footer	<input checked="" type="checkbox"/>
PDF Chart depth output selection	<input checked="" type="checkbox"/>
Selecting page break per level	<input checked="" type="checkbox"/>
Configurable PDF	<input checked="" type="checkbox"/>
Open PDF Directly	<input type="checkbox"/>
Generate Bookmarks	<input checked="" type="checkbox"/>
Page Numbers Format	Simple ▾
Page numbering height	Top ▾
Horizontal position of the page numbering	Right ▾
Upper Margin	<input type="text"/>
Bottom Margin	<input type="text"/>
Right Margin	<input type="text"/>
Left Margin	<input type="text"/>
Print Background	<input checked="" type="checkbox"/>
JS execution time	<input type="text" value="2000"/>
Timeout for chart's image creation	<input type="text" value="150"/>
Displays title below each grouping	<input type="checkbox"/>
Display header in all pages	<input checked="" type="checkbox"/>
Display the title on all pages	<input checked="" type="checkbox"/>

PDF Orientation

Allows you to set whether to print in Portrait or Landscape orientation.

PDF Format

It allows you to define the PDF paper format (letter, A4, others).

Print Type

It allows you to set whether the print mode as colored or economical with no colors.

Records per page

It is used only to horizontal Grids to define the number of records to print per page.

Records per page in the Summary

Lets you define the number of records to print per page in the Summary.

Complete lines until the footer

Complete with (empty) lines until the footer.

PDF Chart depth output selection

It allows the end-user to decide on the fly the depth of the Group By to exports.

Selecting page break per level

It allows the end-user to define the break level to skip to the next page in the generated PDF.

Configurable PDF

It allows the end-user to configure the PDF parameters on the fly.

Generate PDF directly

Opens the generated the PDF without the need to display an intermediate page with a link to download the file.

Generate bookmarks

Generate bookmarks automatically according to the Group By.

Page numbering format

Set the numbering format as Simple (1,2,3,4,5, ...) Full (1 / n, 2 / n, 3 / n, ...) or no numbering.

Page numbering height

It allows you to align the numbering vertically, at the top or bottom of the page.

Horizontal position of page numbering

It allows the horizontal alignment of page numbering (Left, Center, Right).

Margin

Set the PDF margins value in millimeters (Up, Down, Right, Left).

Print background

Allows you to print background in the PDF file.

JS Runtime

The maximum time, in seconds, of server waiting when running JS.

Timeout for chart's image creation

Set the timeout to wait for the chart image creation in the PDF

Displays title below each grouping

Set to display the title of the Group at the beginning of each grouping.

Display header in all pages

It allows displaying the header even if it is not enabled in the grid.

Display the title on all pages

Display the title in all the PDF pages.

WORD Settings

General settings	
ATTRIBUTE	VALUE
Word configurable	<input type="checkbox"/>
Grid Export modules	<input checked="" type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Summary export modules	<input type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Export with password	<input type="checkbox"/>
Select columns	<input type="checkbox"/>
Open WORD Directly	<input type="checkbox"/>

Word configurable

It allows the end-user to configure the export parameters on the fly.

Export Modules of Grid

This option allows the developer to choose which modules are available in the grid export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
---------------------	---

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Summary Export Modules

This option allows the developer to choose which modules are available in the summary export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
---------------------	---

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Export with password

This option allows the developer to set a password to export the file.

Export with password	<input checked="" type="checkbox"/>
Export with password	<input type="text"/>

Password

It allows the developer to set the requested password when exporting the file.

Runtime Example:

PDF settings	
GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Print settings	<input type="text" value="Color"/>
Page Layout	<input type="text" value="A4 (210 X 297 mm)"/>
Orientation	<input type="text" value="Portrait"/>
Other options	<input checked="" type="checkbox"/> Generate Bookmarks <input checked="" type="checkbox"/> Displays the header on all pages <input checked="" type="checkbox"/> Displays the title on all pages
Password	<input type="text" value="....."/>

Select columns

It allows the user has the option to choose the columns he wants to export at runtime.

PDF settings	Select Columns
	<input type="checkbox"/> Accountid <input type="checkbox"/> Accountdescription <input type="checkbox"/> Accountorder

Open WORD Directly

Opens the generated WORD file without the need to display an intermediate page with a link to download it.

WORD Settings

WORD Settings	
ATTRIBUTE	VALUE
Print Type	<input type="text" value="Color"/>
Rows per Page in the Grid	<input type="text" value="0"/>
Rows per Page in Summary	<input type="text" value="0"/>

Print Type

It allows you to set whether the print mode as colored or economical with no colors(Black and White).

Lines per page in Grid

It allows to define the number of records exporting per page in the Grid.

Lines per page in Summary

Lets you define the number of records to print per page in the Summary.

CSV Settings

General settings	
ATTRIBUTE	VALUE
Configurable CSV	<input type="checkbox"/>
Grid Export modules	<input checked="" type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary
Summary export modules	<input type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary
Export with password	<input type="checkbox"/>
Select columns	<input type="checkbox"/>
Open CSV Directly	<input type="checkbox"/>

Configurable CSV

It allows the end-user to configure the CSV export parameters on the fly.

Export Modules of Grid

This option allows the developer to choose which modules are available in the grid export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
---------------------	---

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Summary Export Modules

This option allows the developer to choose which modules are available in the summary export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
---------------------	---

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Export with password

This option allows the developer to set a password to export the file.

Export with password	<input checked="" type="checkbox"/>
Export with password	<input type="text"/>

Password

It allows the developer to set the requested password when exporting the file.

Runtime Example:

PDF settings	
GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Print settings	Color
Page Layout	A4 (210 X 297 mm)
Orientation	Portrait
Other options	<input checked="" type="checkbox"/> Generate Bookmarks <input checked="" type="checkbox"/> Displays the header on all pages <input checked="" type="checkbox"/> Displays the title on all pages
Password

Select columns

It allows the user has the option to choose the columns he wants to export at runtime.

Open CSV Directly

Opens the generated CSV file without the need to display an intermediate page with a link to download it.

CSV Settings

ATTRIBUTE	VALUE
Lines Separator	CRLF
Columns Separator	;
Text Delimiter	"
Add Label	<input type="checkbox"/>

Line separator

It allows you to define the line separator character, each line representing a record.

Column separator

It allows you to define the column separator character, each column representing a database field.

Text delimiter

It allows you to define the character used to delimit the text of the columns when we have database fields of type String.

Add label

It allows you to define displaying the label of the columns in the first line of the file.

XLS Settings

ATTRIBUTE	VALUE
Export with password	<input type="checkbox"/>
Select columns	<input checked="" type="checkbox"/>
Open Excel Directly	<input type="checkbox"/>

Export with password

This option allows the developer to set a password to export the file.

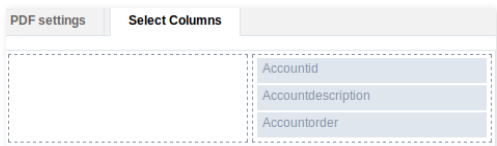
Password

It allows the developer to set the requested password when exporting the file.

Runtime Example:

Select columns

It allows the user has the option to choose the columns he wants to export at runtime.



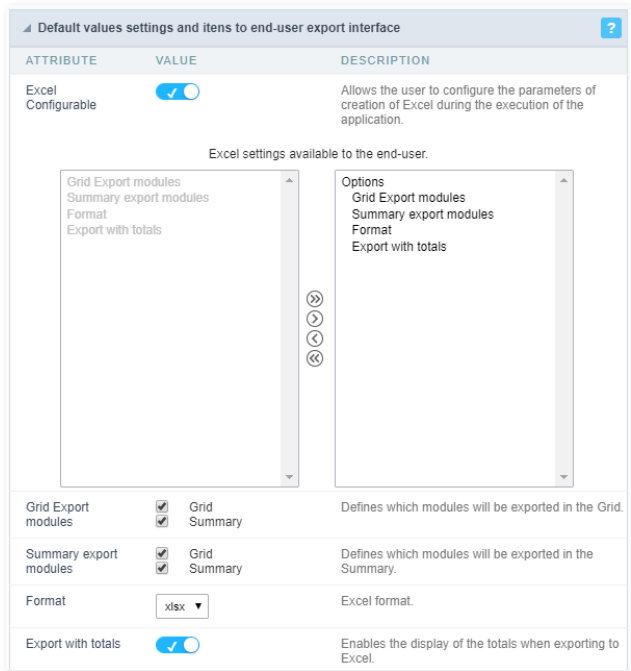
XLS Configurable

It allows the end-user to configure the XLS export parameters on the fly.

Open XLS Directly

It allows the end-user to configure the XLS export parameters on the fly.

XLS Settings



Export Modules of Grid

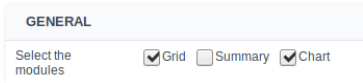
This option allows the developer to choose which modules are available in the grid export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.



You can see the Summary option disabled at running time.



Summary Export Modules

This option allows the developer to choose which modules are available in the summary export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.



You can see the Summary option disabled at running time.



Format

It allows defining the Excel format of the generated document (xls or xlsx).

Export with totals

Show the totals when exporting the application data to Excel.

XML Settings

General settings	
ATTRIBUTE	VALUE
Configurable XML	<input type="checkbox"/>
Grid Export modules	<input checked="" type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary
Summary export modules	<input type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary
Export with password	<input type="checkbox"/>
Select columns	<input type="checkbox"/>
Open XML Directly	<input type="checkbox"/>

Configurable XML

It allows the end-user to configure the XML export parameters on the fly.

Export Modules of Grid

This option allows the developer to choose which modules are available in the grid export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
---------------------	---

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Summary Export Modules

This option allows the developer to choose which modules are available in the summary export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
---------------------	---

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Export with password

This option allows the developer to set a password to export the file.

Export with password	<input checked="" type="checkbox"/>
Export with password	<input type="text"/>

Password

It allows the developer to set the requested password when exporting the file.

Runtime Example:

PDF settings

GENERAL

Select the modules Grid Summary Chart

Print settings

Page Layout

Orientation

Other options Generate Bookmarks
 Displays the header on all pages
 Displays the title on all pages

Password

Select columns

It allows the user has the option to choose the columns he wants to export at runtime.

PDF settings **Select Columns**

Accountid
Accountdescription
Accountorder

Open XML Directly

Opens the generated XML file without the need to display an intermediate page with a link to download it.

XML Settings

XML

ATTRIBUTE	VALUE
Use Label on the Tag	<input type="checkbox"/>
XML Format	<input type="radio"/> Attr <input checked="" type="radio"/> Tag

Use Label on the Tag

It allows using the field labels on Tags of the XML file.

XML Format

Define how to generate the records in the XML file, if it will store the values on attributes or new elements. (Attr or Tag)

Print Settings

General settings

ATTRIBUTE	VALUE
Configurable Print HTML	<input type="checkbox"/>
Grid Export modules	<input checked="" type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Summary export modules	<input type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Select columns	<input type="checkbox"/>

Configurable Print HTML

It allows the end-user to configure the HTML Printing parameters on the fly.

Export Modules of Grid

This option allows the developer to choose which modules are available in the grid export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules Grid
 Summary
 Chart

You can see the Summary option disabled at running time.

GENERAL

Select the modules Grid Summary Chart

Summary Export Modules

This option allows the developer to choose which modules are available in the summary export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

You can see the Summary option disabled at running time.

Export with password

This option allows the developer to set a password to export the file.

Password

It allows the developer to set the requested password when exporting the file.

Runtime Example:

Select columns

It allows the user has the option to choose the columns he wants to export at runtime.

Print Settings

ATTRIBUTE	VALUE
Print Mode	Both
Print Type	Both
Rows per Page	0
Rows per Page in Summary	0
Print Background	<input checked="" type="checkbox"/>

Print Mode

It allows choosing the content to print. (Current Page or Full Report)

Print Type

Define the file print mode. (Black & White or Colored)

Rows per Page

It defines how many Grid rows to display per page in the file.

Rows per Page in Summary

It defines how many Summary rows to display per page in the file.

Print Background

Allows displaying the background in an HTML form.

RTF Settings

General settings	
ATTRIBUTE	VALUE
Select columns	<input type="checkbox"/>
Open RTF Directly	<input type="checkbox"/>

Select columns

This option allows the end-user to select which fields he wants to export to the file.

PDF settings	Select Columns						
	<table border="1"><tbody><tr><td><input type="checkbox"/></td><td>Accountid</td></tr><tr><td><input type="checkbox"/></td><td>Accountdescription</td></tr><tr><td><input type="checkbox"/></td><td>Accountorder</td></tr></tbody></table>	<input type="checkbox"/>	Accountid	<input type="checkbox"/>	Accountdescription	<input type="checkbox"/>	Accountorder
<input type="checkbox"/>	Accountid						
<input type="checkbox"/>	Accountdescription						
<input type="checkbox"/>	Accountorder						

Open RTF Directly

Opens the generated RTF file without the need to display an intermediate page with a link to download it.

SQL Settings

This interface allows configuring the related database settings, such as the SQL statement, the used database connection, case sensitive, and others.

ATTRIBUTE	VALUE	DESCRIPTION
SQL Select Statement	<pre>SELECT customerid, companyname, contactname, contacttitle, birthdate, country, regionid, stateid, city,</pre>	
Limit	<input type="text"/>	It sets the number of records to be retrieved from the SQL statement.
SQL Preparation	<input type="text"/>	
Connection	<input type="text" value="conn_example"/>	Connection name to access the database.
Use Customized Message	<input type="checkbox"/>	Use a customized error message when the application has no records.
No Records Message	<input type="text"/>	When the application has no records, it will display this customized text.
Font	<input type="text" value="Aa"/>	Font face of the error message.
Font Size	<input type="text" value="12"/>	Font size of the error message.
Font Color	<input type="text" value="#000000"/>	Font color of the error message.
Variable for Table	<input type="text"/>	Variable name used for replacing the table name. Please indicate the name of the table that will be replaced by the variable value.
Fields Variables	<div style="border: 1px solid black; padding: 5px;"> <p>Variable</p> <input type="text"/> <p>customerid</p> </div>	Variables for substitution of the field names on the application. For each dynamically determined field, inform the name of the variable and the field that will be substituted.
Case Sensitive	<input checked="" type="checkbox"/>	Use case sensitive.

Grid SQL configuration

SQL Select Statement

It allows you to define the primary SQL of the application. You can edit this SQL to add or delete fields.

Limit

Lets you limit the display in the number of records retrieved by SQL query.

SQL Preparation

You can enter SQL commands or procedure names to execute them before the primary SQL of the application.

Connection

It allows defining the database connection of the application. You can change the connection to another one that has the same table.

Use Customized Message

Lets you define to display the "no records" message or not.

No Records Message

Lets you set the message when the application has no records.

Font

This option is available when using the "Use Customized Message" option. It lets you set the font for the message.

Font Size

This option is available when using the "Use Customized Message" option. It lets you to set the font size.

Font Color

This option is available when using the "Use Customized Message" option. It lets you to set the font color.

Variable for Table

It allows to use a variable to change a part of the string containing the table name.

Fill the first input with the name of the variable (replace to). The second input you should fill with the part of the name of the table you want to replace (replace from).

Fields Variables

It allows to use a variable to change a part of the string containing the field name.

Fill the first input with the name of the variable (replace to). The second input you should fill with the part of the name of the field you want to replace (replace from).

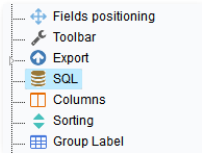
Case sensitive

It defines if the database connection uses case sensitive or not.

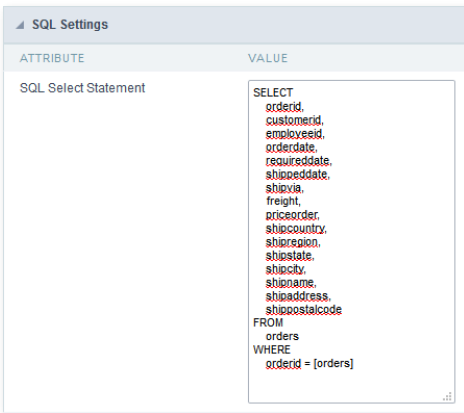
Creating Parameters for Applications Links

Parameters are used to pass values between applications. When the target application is a query, we recommend adding a global variable in the WHERE clause to filter the records that meet a specific condition.

To configure the parameter, in the destination application, access the SQL option in the left side menu.



We must edit the SQL of the grid application, adding the WHERE clause and the field that receive the value through a global variable(as shown in the image below).

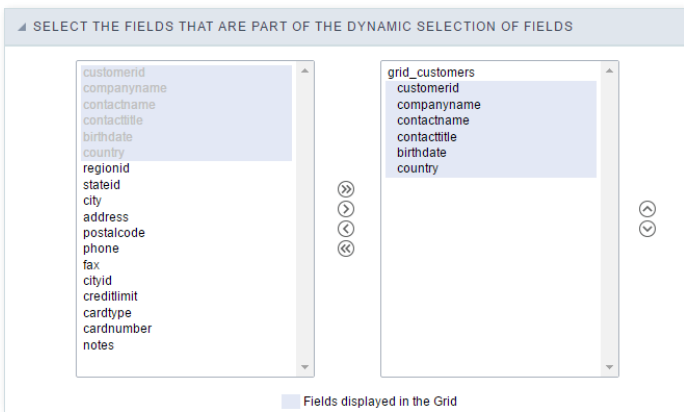


after edition, we must generate the source code of the application clicking on the button -



- or using the hotkey **F8**.

You can display or hide fields dynamically inside the Grid application using the button "Columns". This option is used to select the fields/columns that will be part of this dynamic fields selection.



Grid columns' settings

The box in the left represents the fields that are NOT being displayed within the dynamic fields selection.

The box in the right represents the fields that are being displayed within the dynamic fields selection.

Field Settings

Select the fields that allow sorting.

On/Off

All

None

- *customerid
- *companyname
- *contactname
- *contacttitle
- *birthdate
- *country
- regionid
- stateid
- city
- address

Sort Fields			
customerid	Ascendant	companyname	Ascendant
contactname	Ascendant	contacttitle	Ascendant
birthdate	Ascendant	country	Ascendant
regionid	Ascendant	stateid	Ascendant
city	Ascendant	address	Ascendant
postalcode	Ascendant	phone	Ascendant
fax	Ascendant	cityid	Ascendant
creditlimit	Ascendant	cardtype	Ascendant
cardnumber	Ascendant	notes	Ascendant

Advanced sorting

ATTRIBUTE	VALUE	DESCRIPTION
Sorting fields	<input checked="" type="radio"/> Use displayed fields <input type="radio"/> Use columns <input type="radio"/> Define Columns	Default for fields that will be displayed in the advanced sorting
Display icon only on mouseover	<input checked="" type="checkbox"/>	Displays the sorting icon only when the mouse is on the label of the field.

This property defines which fields allow the end-user to sort on the fly, just by clicking on the column (field) he wants to sort by. When sorting the fields, it preserves the GroupBys ordering by default. Also, Scriptcase enables the sorting of the first six application fields, but you can enable it for all fields.

On/Off

All

None

- *customerid
- *companyname
- *contactname
- *contacttitle
- *birthdate
- *country
- regionid
- stateid
- city
- address

Sort Fields

It defines the sorting for each field of the application. You can set them as "Ascendant" or "Descendant".

Sort Fields			
customerid	Ascendant	companyname	Ascendant
contactname	Ascendant	contacttitle	Ascendant
birthdate	Descendant	country	Ascendant
regionid	Descendant	stateid	Ascendant
city	Ascendant	address	Ascendant
postalcode	Ascendant	phone	Ascendant
fax	Ascendant	cityid	Descendant
creditlimit	Descendant	cardtype	Ascendant
cardnumber	Ascendant	notes	Ascendant

Advanced Sorting

Define the fields that are available when clicking on the "Sorting" button of the Grid.

ADVANCED SORTING	
ATTRIBUTE	VALUE
Sorting fields	<input checked="" type="radio"/> Use displayed fields <input type="radio"/> Use columns <input type="radio"/> Define Columns

Use displayed fields

Defines that all the fields are available when clicking on the "Sorting" button of the application.

Use columns

Defines that all the fields set in the "Columns" option are available when clicking on the "Sorting" button.

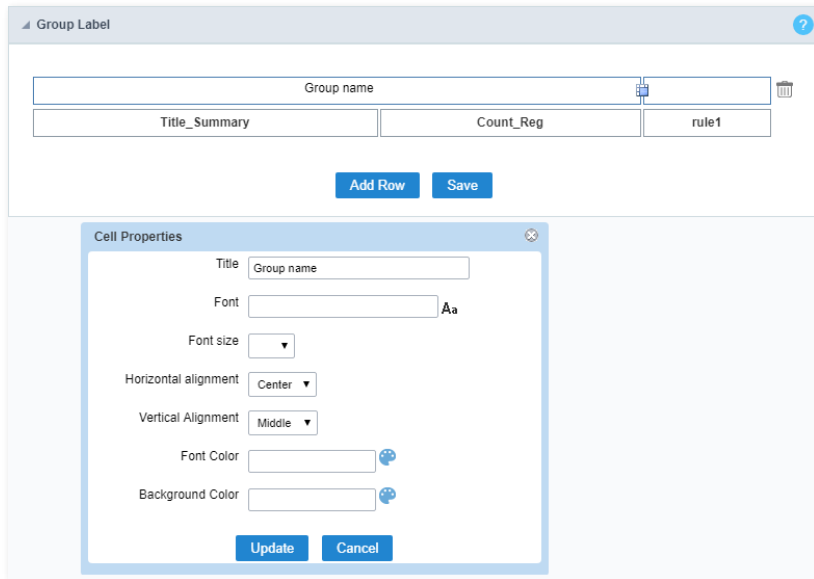
Define Columns

Allows selecting the fields freely.

ADVANCED SORTING	
ATTRIBUTE	VALUE
Sorting fields	<input type="radio"/> Use displayed fields <input type="radio"/> Use columns <input checked="" type="radio"/> Define Columns
Defined fields for the sorting	<div style="display: flex; align-items: center;"><div style="border: 1px solid #ccc; padding: 2px; width: 150px;"><ul style="list-style-type: none">customeridcompanynamecontactnamecontacttitlebirthdatecountryregionidstateid</div><div style="margin: 0 10px;"><input type="button" value="Add >>"/> <input type="button" value="Remove <<"/></div><div style="border: 1px solid #ccc; width: 50px; height: 40px; margin-left: 10px;"></div></div>

With this option, you can insert one or more titles for the columns in the application, allowing to modify its size, color, font, and position.

The Group Label doesn't work with a dynamic display of fields.



Editing Group Label cell properties

Title

It allows you to define a Title for the Group Label.

Font

Set the font family for the group label title.

Font Size

Set the font size for the group label title.

Horizontal Alignment

Define the horizontal alignment of the group label title. Left, Center, or Right.

Vertical Alignment

Define the vertical alignment of the group label title. Middle, Bottom, or Top.

Font Color

Set the font color for the group label title.

Background Color

Define a background color for the group label title.

Security ?		
ATTRIBUTE	VALUE	DESCRIPTION
Use Security	<input type="checkbox"/>	Application needs validation to run. Read about the sc_apl_status macro.
Url output of the security	<input type="text"/>	Uri to be redirected if the user does not have access to this application.
Use Password	<input type="checkbox"/>	Enter password required to run the application.
Request password just once	<input type="checkbox"/>	Request password just once per session without asking for password every time you open the application in the same session.
Enable direct call by URL	<input checked="" type="checkbox"/>	Allow an application to be called by typing the URL directly in the browsers.

Application Security Configuration

Use Security

Turns the application accessible just when using the macro sc_apl_status or using the security module. When it is enabled, the application displays a message "User not Authorized" if the user tries to access it directly.

Url output of the security

It redirects the application to another URL after displays the warning "User not Authorized".

Use Password

Turns the application accessible just by using a password.

Request password just once

Request the password once per session, not needing to inform the password every time you access the application.

Enable direct call by URL

Allows to call the application directly from the URL in the browsers.

Allows calling the application directly from the URL in the browsers. In the development environment, you can disable those options for testing, it avoids requiring the login every time you run the application. To disable the security and password in development, you must access Options > My ScriptCase and uncheck the options: Enable Use of Security, Enable Use of Password.

This interface allows you to define a Log schema to the app. The Log schema tracks what the end-users are doing in the app. Those are the events it can tracks: Access, Insert, Update, and delete.

To learn how to create a Log schema [see here](#).

The screenshot shows a configuration window titled "LOG". It features a table with two columns: "ATTRIBUTE" and "VALUE". The "Schema" attribute has a dropdown menu currently set to "log". Below the table, there are two empty list boxes: "Events" on the left and "Access" on the right. Between these two boxes are two blue buttons: "Add >>" and "Remove <<".

Application Log configuration

Schema Select an existing log schema for the application.

Events You can select just specific events to save in the log.

GANTT CHART SETTINGS	
ATTRIBUTE	VALUE
Output Format	HTML5 ▼
Field label	▼
Width	600
Height	450
Months	12
Start Date	▼
End Date	▼
Display format	Regional settings ▼
Percent Completed	▼
Resource	▼

Gantt Chart settings

A Gantt chart is a type of bar chart that illustrates a project schedule. Gantt charts illustrate the start and end dates of the terminal elements and summary elements of a project. Terminal elements and summary elements comprise the work breakdown structure of the project.

Output Format	Output format, you can choose of HTML or HTML5 (the default option is HTML5)
Field label	Field containing the record label of the Gantt chart. You can choose which field will appear within the left column to describe the tasks.
Width	Gantt chart width. The default unit for this width is pixels, however you can also apply it in percent by adding the “%” sign after the number (i.e: 100%).
Height	Gantt chart height. The default unit for this width is pixels, however you can also apply it in percent by adding the “%” sign after the number (i.e: 100%).
Months	Allows you to set the number of months that will be displayed in the Gantt chart.
Start Date	Using this combo box you can choose the field that stores the start date, you need to have this record within the connected table.
End Date	Using this combo box you can choose field that stores the end date, you need to have this record within the connected table.
Display format	Allows you to set the display format of the fields date type.
Percent Completed	Using this combo box you can choose the field that stores the information regarding the task completion (percentage), you need to have this record within the connected table.
Resource	Field containing the name of the resource allocated to the task.

This feature allows the user to save the current filter state. In this option the user can create rules so that the current filter can be used in other Grid applications. This way you can configure different filters without the need to filter again.

The screenshot shows a dialog box titled "SAVE CRITERIA". It has two checked checkboxes: "Enable Public" and "Use Rules". Below "Enable Public" is a text field containing "(lang_srch_public)". Below "Use Rules" is a vertical list with three items: "New rule", "Edit", and "Delete". Each item has a small circular icon to its left. The "New rule" button is highlighted with a red rectangle.

Enable Public Displays the Save button within the Grid application

The screenshot shows a dialog box titled "SAVE CRITERIA". It has two text input fields: "Name" and "Variable name". The "Name" field contains "[var_login]" and is highlighted with a red rectangle. Below the "Name" field is a "Label" field. To the right of the "Name" field is a vertical list with three items: "Add", "Delete", and "Finish". Each item has a small circular icon to its left. The "Add" button is highlighted with a red rectangle. Below the "Label" field are "Finish" and "Cancel" buttons.

Use Rules Using this option you can enable more rules for the save button

1. [Home](#)
2. Applications

General Overview

The applications can have two types of fields.

Table Fields

Columns of the table connected to the application and their data types.

Example:

In a table with the following structure:

```
CREATE TABLE `orders` (
  `orderid` int(11) NOT NULL,
  `customerid` varchar(5) NULL DEFAULT NULL,
  `employeeid` int(11) NULL DEFAULT NULL,
  `orderdate` date NULL DEFAULT NULL,
  `requireddate` date NULL DEFAULT NULL,
  PRIMARY KEY (`orderid`) USING BTREE
);
```

These are the listed fields on left menu **Fields**:

Virtual Fields

Fields created inside application to assist in the development process.

How to create a virtual field

1. Inside an application, access the left menu **Fields** and click on **New Field** option.
2. Choose how many fields do you want to create:

1. Create the fields and define type, name and label for each one:

- **Type:** Data type of the field.
- **Name:** Internal name of the field. Used to identify the field on events and Scriptcase interface.
- **Label:** Title of the field displayed on the application.

Scriptcase **doesn't insert** virtual fields on the table after its creation.

Types of fields by applications

Application Table Fields Virtual Fields

Grid	<input type="text"/>	<input type="text"/>
Procedure	<input type="text"/>	<input type="text"/>
Chart	<input type="text"/>	<input type="text"/>
Form	<input type="text"/>	<input type="text"/>
Control	<input type="text"/>	<input type="text"/>
Search	<input type="text"/>	<input type="text"/>
PDF Report	<input type="text"/>	<input type="text"/>
Calendar	<input type="text"/>	<input type="text"/>

The applications **Menu**, **Tree Menu**, **Dashboard** and **Blank** were not listed because they haven't manage fields.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Text

Text

General Settings

This type of field allows the developer to create quickly fields to display data from the database, where the final user can see the data in the way it was set by the developer.

Type of Data

Defines the type of the field in the application. In this case we should select **Text**.

Label

Defines the title that will be displayed in the field when the application is executed. The terminology used in the interface is fundamental to the system have a good usability, when should use names and familiar terms to the final user of the application, instead of using terms from the system.

For example, this text field that has the name **customerid**, the client would have a much better understanding of the functionality of the field when we define the label as **Customer Name**.

Not only a fixed text, the **Label** attribute allows the use of langs to define the field's title, making it possible to use your application in a multi language project.

Case Settings

This option allows the developer to inform how the text will be displayed in runtime.

The options available are:

Lower Case: Every letter in the text will be converted to lower case.

Upper Case: Every letter of the text will be converted to upper case.

Capitalize first word: The first letter of the first world will be converted to upper case.

Capitalize all words: The first letter of every words will be converted to upper case.

Show HTML Content

When this option is active every HTML, CSS and JavaScript content that are in the database will be displayed with the main value.

Field Mask

Defines the field mask. There are two types of mask described in the table below:

Character Description

- X It is overridden by any characters returned by the database. Filled is required and the value will be completed with leading zeros when there are fewer characters than the size of the mask.
- Z It is overridden by any characters returned by the database. Its filling is optional and when there are fewer characters than the mask size, nothing will be done about the characters that are missing. In addition the leading zeros will be suppressed. When used in conjunction with the X necessarily the use of this character is to the left of the mask.
- 9 It represents any numeric character (from 0-9)
- A It represents an alpha numeric character (A-Z,a-z)
- * It represents any alpha-numeric character (A-Z,a-z,0-9)entered by the user.

For example, it is possible to set the mast to display a telephone number:

It will be show with this format on runtime:

It is also possible to set the field mask like those examples:

Field mask examples:

Telephone number

Field	Mask Input	Typed Value	Formatted Value
Phone Number	+99 99 9999 - 9999	123456789012	+12 34 5678 - 9012
Phone Number	(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
Phone Number	(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
Phone Number	(zz) xxxx - xxxx	12345678	() 1234 - 5678
Software Key (Letters only)	AAAA-AAAA-AAAA-AAAA	QWERTYUIASDFGHJK	QWER-TYUI-ASDF-GHJK
Software Key (Letters and numbers)	xxxx-xxxx-xxxx-xxxx	Q1W2E3R4T5Y6U7I8	Q1W2-E3R4-T5Y6-U7I8
License Plate	AAA - 9999	QWE1234	QWE - 1234
Scriptcase Serial Key	A999A999A999-**	D111H111G111DG2P	D111H111G111-DG2P
Multiple masks (Phone Number)	9999-9999;(99)9999-9999; 9999 999 9999		+99 99 9999-9999

Repeat Value:

When this option is active the value of the field will be repeated if the previous database register is the same.

Example:

SQL Type

Informs the type of the field in the database.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced by the curly brackets {}. While running the application, the field within the curly brackets is replaced by its value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
- **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
- **Summary** : Applies the lookup only for the summary (HTML and PDF).
- **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Martial Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

PDF Configuration


Line Break

This option allows breaking the line in the PDF file when exported.

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

 *Bar Chart Interface.*

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

 *Chart layout settings interface.*

1. [Home](#)
2. [Applications](#)
3. Grid
4. Multiple Lines Text

Multiple Lines Text

General Settings

Multiple Lines Text field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Multiple Lines Text, it accepts letters, numbers and special characters in multiple lines.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Case Settings** : Convert the letter from the field when losing focus. The options are:
 - **Upper case** : All in Upper Case
 - **Lower case** : All in Lower case
 - **Capitalize first word** : Capitalizes the first letter of the first word
 - **Capitalize all words** : Capitalizes the first letter of all the the words
- **Show HTML content** : Determines if the HTML contained in the field will be displayed or not. If enabled, the HTML will be displayed, otherwise the HTML will be interpreted by the browser.
- **Grid Mask** : Defines the mask for the field display. There are two typed of masks described below:

Character	Description
X	Placeholder to any character. Replaced by any character. If number of characters entered are less then the mask size, the field value is completed with zeros (Filling full size field entry is required).
Z	Replaced by any character retrieved from database. Suppress zeros at field left (Complete field filling is optional). When used combined with the mask character X it should be placed at the mask left.

Mask Examples

Mask	Field Value	Formatted Value
(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
(zz) xxxx - xxxx	1234567890	(12) 3456 - 7890
(zz) xxxx - xxxx	12345678	() 1234 - 5678
(zz) xxxx - xxxx	0012345678	() 1234 - 5678

- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.

- **Run content in JavaScript** : If enabled, the JavaScript will be interpreted by the browser, otherwise the JavaScript will be displayed.
- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

PDF Configuration

Line Break

This option allows breaking the line in the PDF file when exported.

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.

- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

Chart layout settings interface.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Integer

Integer

On this page, you will learn how you can configure settings related to the Number field. From the use of specific symbols display to the mode in which they are displayed. And thus, boost the application.

Data type:

It sets the application field type. When the field is set as Number, it is permissible to define formatting rules of integers.

Label:

It sets the title that will be displayed in the field when you run the application. The terminology used in the interface is fundamental for your system has a good usability. In this case, we recommend you to use names and terms familiar to the end user of the application, instead of using terms originated in the system.

Field Mask Input:

It sets the mask input for the field. There are two types of mask as described in the table below:

Character Description

X	It is overridden by any characters returned by the database. Filled is required and the value will be completed with leading zeros when there are fewer characters than the size of the mask.
Z	It is overridden by any characters returned by the database. Its filling is optional and when there are fewer characters than the mask size, nothing will be done about the characters that are missing. In addition the leading zeros will be suppressed. When used in conjunction with the X necessarily the use of this character is to the left of the mask.
9	It represents any numeric character (from 0-9)
A	It represents an alpha numeric character (A-Z,a-z)
*	It represents any alpha-numeric character (A-Z,a-z,0-9)entered by the user.

Mask Input Examples:

Field	Mask Input	Typed Value	Formatted Value
Phone Number	+99 99 9999 - 9999	123456789012	+12 34 5678 - 9012
Phone Number	(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
Phone Number	(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
Phone Number	(zz) xxxx - xxxx	12345678	() 1234 - 5678
Software Key (Letters only)	AAAA-AAAA-AAAA-AAAA	QWERTYUIASDFGHJK	QWER-TYUI-ASDF-GHJK
Software Key (Letters and numbers)	xxxx-xxxx-xxxx-xxxx	Q1W2E3R4T5Y6U7I8	Q1W2-E3R4-T5Y6-U7I8
License Plate	AAA - 9999	QWE1234	QWE - 1234
Scriptcase Serial Key	A999A999A999-**	D111H111G111DG2P	D111H111G111-DG2P
Multiple masks (Phone Number)	9999-9999;(99)9999-9999; 9999 999 9999		+99 99 9999-9999

Repeat value:

This option when enabled will allows you to repeat the field value if it is equal to the value of the previous record in the database.

Example:

Tipo SQL:

Reports the SQL type of the field as it is configured in the database.

Use regional settings:

It allows you to apply regional settings to the number fields formatting. When this option is not selected it will be displayed the attributes group, minus sign and negative number format.

Grouping:

It allows you to define which character is used to separate thousands. This option is only available when the option to use regional settings is disabled.

Negative sign:

It allows you to define which character will be used for the display of negative numbers. This option is only available when the option to use regional settings is disabled.

Negative number format:

It allows you to define the placement of the negative sign in relation to value. This option is only available when the option to use regional settings is disabled.

Color for the negative values:

It allows you to define a color when the value is negative, improving the understanding of the end user about that kind of value.

Example:

Display the value in words:

The value of the field will be displayed in full on application. This feature can facilitate the comprehension and understanding of the user.

Example:

Line size:

Maximum size in characters to be displayed in the value cell, in full. When this value is exceeded the line will break within the cell.

Lookup Settings

The Lookup is a way used to provide the end user a list of values description/name where the items can be selected and used in the application. For example, for a field sex that in the database is represented with M or F values can be presented as male or female dynamically with the lookup. So you will be able to retrieve these values from a database table or manually, if the reference table does not exist.

Example:

Lookup Methods

There are two lookup methods available:

Manual: To use this method it is necessary to inform the selection conditions manually to the Lookup.

Automatic: To use this method it is necessary to use a lookup query from the database to access the reference table.

Editing-automatic lookup:

Select Command:

It sets the SQL command that will retrieve the value to display inside the Grid field. The command must have the following format

```
SELECT field_to_be_displayed FROM table WHERE key_field = {grid_field}
```

The Grid field must be always referenced between curly braces {}, So at run time, the field braces will be replaced by its value in the table.

Multiple Options:

When the Select command informed return multiple values, the option must be selected.

Delimiter:

It sets the tab of values returned by the Select command, this option should be filled when the "Yes" option is checked in the multiple attribute options.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

Use a specific connection:

This option allows you to select another existing connection in the project, other than the current connection, to select the values for the lookup.

Edit Lookup - Manual:

Lookup used to modify the display of the field with predefined values.

Lookup Types

It sets the operating mode of the lookup can be: single value, multiple values (delimiter), multiple values (position) and multiple values (binary).

Single value: it will be displayed only one value for the field. There is a direct replacement of the value stored in the Bank for a label defined in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value of "M" will be replaced by "Male".

Multiple Values (Delimiter):

By using this option it will be displayed multiple values for the selected field. The values must be separated by a delimiter to be informed. A replacement of parts from a string stored in the field, separated by a delimiter for values contained in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value S;M will be replaced by Sports and Movies.

Delimiter: Character used to separate the values within the same string.

Multiple Values (Position)

Allows you to recover information stored in a single string of the selected field. In order for this information to be retrieved must be informed, in addition to the label, the starting position and the number of bytes that each information occupies within the string.

As an example we will use a string to store Sex, Marital status and Hobby respectively. Sex occupies one byte, Status one byte and Hobby two bytes .

For this purpose we define the list as:

Label	Value	Start	Size
Male	M	1	1
Female	F	1	1
Married	M	2	1
Single	S	2	1
Study	ST	3	2
Sportes	SP	3	2
Reading	LE	3	2

Example: the string MMST would be presented in the query as: Male Married Study

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value M will be replaced by Male.

Start: Starting position in the string where the information is recorded. The first position is always 1.

Size: Number of bytes that the information occupies in the string.

Multiple values (binary):

It allows you to retrieve several information stored in decimal form of the selected field.

As examples we will use the following list (although not informed in the interface of inclusion of values in the list, each item has a value assigned automatically according to the order):

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

To display the data, a decomposition of the decimal number stored in the bank is performed. For example, numbers 11 and 12 (written in the database) would be decomposed as follows:

$$11 = 1 + 2 + 8 = (\text{Sports} - \text{Culture} - \text{Reading})$$

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

$$12 = 4 + 8 = (\text{Leisure} - \text{Reading})$$

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

Label:

Text that will be displayed in the Grid.

Include Button:

Includes in the list the values populated in the Label and value fields.

Alter Button:

It changes the attributes of the selected item.

Delete Button:

Deletes the selected item.

Clean Button:

It cleans the fields

Save Button:

Allows the user to save the entire list, for later use using the Load Lookup definition option.

Load Lookup definition:

Allows you to load a predefined value list for use as a lookup.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

• Field Settings

- **Font Style** : Allows to select the font family, that will be applied to the application field.
- **Font Size** : Allows to choose the font size, that will be applied to the application field.
- **Italic Text** : Formatting the Text to be Italic.
- **Bold Text** : Formatting the Text to be Bold.
- **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
- **Number of characters** : Amount of characters in bytes to display.
- **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
- **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Background Image** : Background Image of the field.
- **Line Break** : Allows the Line Break.
- **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
- **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
- **Width** : Allows to define a width to the field.
- **Height** : Allows to define a height to the field.
- **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
- **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
- **Bold** : Applies the bold type font to the field.

Line Chart

Using this guide, you can define ac chart for each line.

- **Line Chart Type** : There are two types of charts that can be used. Below you can see their description:
- **Bar Chart** : This option allows to display a bar chart relating to the column that you are working with. The size of the bars is done by math relating to the Grand Total of the column. The value informed in the width of the chart defines the size of the images that will be displayed.

In the example shown above, the width of the Chart is 200 pixels.

- **Line Chart** : This option allows to display a Line Chart relating to the column that you are working with . The amount of lines are done by dividing the value of a column by the value informed to the number of icons.

In the example above, the field Number of Icons, it was informed the value 1000, so the number of stars besides, it determined by dividing the displayed value by 1000.

- **Display Value** : It will display the value and the chart at the same time, the value being on the left or right side of the bar or lines.
- **Number of Icons** : Determines the amount of icons (figures) that will appear for each value on this field. This value will be presented by the dividing the value of the column by the Number of Icons.
- **Chart width** : Width of the chart in pixels. This field is used by the option Bar Chart. In case it is not informed, the value used is 200 pixels.
- **Chart height** : Height of the chart in pixels. This field is used by the option Bar Chart.
- **Margin** : Charts margin. This option is only used by the bar chart.
- **Image / Positive Color** : The way that it is informed will variate depending on the options below:
 - **Bar Chart** : Inform a color for the field that will be used to form the chart image. Click the icon beside to choose the color.
 - **Line Chart** : Inform an image name for the field that will be used like an icon. Click the icon beside to choose the image "Icon".
- **Image / Negative Color** : The information passed to this field follows the same criteria as the Image / Positive Color field. This parameter will be used in the case the value is negative.

PDF Configuration

Line Break

This option allows breaking the line in the PDF file when exported.

Charts Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

Chart layout settings interface.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Decimal

Decimal

General Settings

Decimal field Configuration Interface.

- **Data Type** : Select the type of field for the application. When it is defined as a Decimal you can define the format of a decimal number.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Accumulated** : When enabled, define that the field will be accumulated with the values of another field. This option is only available on virtual fields.
- **Accumulator field** : Only available when the option "Accumulated" is enabled. This configuration defines which field will be used to accumulate the values. The chosen fields needs to be Integer, Currency or Decimal.
- **Grid Mask** : Defines the mask for the field display. There are two typed of masks described below:

Character	Description
X	Placeholder to any character. Replaced by any character. If number of characters entered are less then the mask size, the field value is completed with zeros (Filling full size field entry is required).
Z	Replaced by any character retrieved from database. Suppress zeros at field left (Complete field filling is optional). When used combined with the mask character X it should be placed at the mask left.

Mask Examples

Mask	Field Value	Formatted Value
(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
(zz) xxxx - xxxx	1234567890	(12) 3456 - 7890
(zz) xxxx - xxxx	12345678	() 1234 - 5678
(zz) xxxx - xxxx	0012345678	() 1234 - 5678

- **Do Not Repeat Value** : Does not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Values format

Format of Values with Regional Settings.

- **Regional Settings** : Applies the Regional Settings to format the number of the fields. When not enabled, you will view attributes grouped, Negative Symbol and negative number format.
- **Digit Grouping Separator** : Defines the character that will be used to separate the thousand.
- **Decimal Separator** : Defines the character that will be used to separate the decimal.
- **Negative sign** : Defines the character that will be used to display negative numbers.
- **Negative number format** : Defines the position of the Negative sign em relation to the value.
- **Color of Negative** : Defines a color for the field when the value is negative.
- **Decimal Precision** : Number of decimal places. This value is part of the field size.
- **Complete with Zeros** : Defines the decimal places that will or not be completed zeros to the right. Example: The value is 125,50, but the decimal precision is 3. If the option is enabled, then the value will 125,500.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

Attribute Value Lookup Description

1 (Sports) + 4 (Pleasure) + 8 (Reading) =13

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

Line Chart

Using this guide, you can define ac chart for each line.

- **Line Chart Type** : There are two types of charts that can be used. Below you can see their description:
- **Bar Chart** : This option allows to display a bar chart relating to the column that you are working with. The size of the bars is done by math relating to the Grand Total of the column. The value informed in the width of the chart defines the size of the images that will be displayed.

In the example shown above, the width of the Chart is 200 pixels.

- **Line Chart** : This option allows to display a Line Chart relating to the column that you are working with . The amount of lines are done by dividing the value of a column by the value informed to the number of icons.

In the example above, the field Number of Icons, it was informed the value 1000, so the number of stars besides, it determined by dividing the displayed value by 1000.

- **Display Value** : It will display the value and the chart at the same time, the value being on the left or right side of the bar or lines.
- **Number of Icons** : Determines the amount of icons (figures) that will appear for each value on this field. This value will be presented by the dividing the value of the column by the Number of Icons.
- **Chart width** : Width of the chart in pixels. This field is used by the option Bar Chart. In case it is not informed, the value used is 200 pixels.
- **Chart height** : Height of the chart in pixels. This field is used by the option Bar Chart.
- **Margin** : Charts margin. This option is only used by the bar chart.
- **Image / Positive Color** : The way that it is informed will variate depending on the options below:
 - **Bar Chart** : Inform a color for the field that will be used to form the chart image. Click the icon beside to choose the color.
 - **Line Chart** : Inform an image name for the field that will be used like an icon. Click the icon beside to choose the image "Icon".
- **Image / Negative Color** : The information passed to this field follows the same criteria as the Image / Positive Color field. This parameter will be used in the case the value is negative.

PDF Configuration

Line Break

This option allows breaking the line in the PDF file when exported.

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

 *Bar Chart Interface.*

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

 *Chart layout settings interface.*

1. [Home](#)
2. [Applications](#)
3. Grid
4. Percent

Percent

General Settings

Percentage field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Percentage, it is allowed to define the format of a percentage.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Accumulated** : When enables, define that the field will be accumulated with the values of another field. This option is only available on virtual fields.
- **Accumulator field** : Only available when the option "Accumulated" is enabled. This configuration defines which field will be used to accumulate the values. The chosen fields needs to be Integer, Currency or Decimal.
- **Grid Mask** : Defines the mask for the field display. There are two typed of masks described below:

Character	Description
X	Placeholder to any character. Replaced by any character. If number of characters entered are less then the mask size, the field value is completed with zeros (Filling full size field entry is required).
Z	Replaced by any character retrieved from database. Suppress zeros at field left (Complete field filling is optional). When used combined with the mask character X it should be placed at the mask left.

Mask Examples

Mask	Field Value	Formatted Value
(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
(zz) xxxx - xxxx	1234567890	(12) 3456 - 7890
(zz) xxxx - xxxx	12345678	() 1234 - 5678
(zz) xxxx - xxxx	0012345678	() 1234 - 5678

- **Do Not Repeat Value** : Don't repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Applies the Regional Settings to format the number of the fields. When not enabled, you will view attributes grouped, Negative Symbol and negative number format.
- **Digit Grouping Separator** : Defines the character that will be used to separate the thousand.
- **Decimal Separator** : Defines the character that will be used to separate the decimal.
- **Negative sign** : Defines the character that will be used to display negative numbers.
- **Negative number format** : Defines the position of the Negative sign em relation to the value.
- **Color of Negative** : Defines a color for the field when the value is negative.
- **Decimal Precision** : Number of decimal places. This value is part of the field size.
- **Complete with Zeros** : Defines the decimal places that will or not be completed zeros to the right. Example: The value is 125,50, but the decimal precision is 3. If the option is enabled, then the value will 125,500.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Methods

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

Lookup Method - Automatic

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

Attribute Value Lookup Description

1 (Sports) + 4 (Pleasure) + 8 (Reading) =13

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

Line Chart

Using this guide, you can define ac chart for each line.

- **Line Chart Type** : There are two types of charts that can be used. Below you can see their description:
- **Bar Chart** : This option allows to display a bar chart relating to the column that you are working with. The size of the bars is done by math relating to the Grand Total of the column. The value informed in the width of the chart defines the size of the images that will be displayed.

In the example shown above, the width of the Chart is 200 pixels.

- **Line Chart** : This option allows to display a Line Chart relating to the column that you are working with . The amount of lines are done by dividing the value of a column by the value informed to the number of icons.

In the example above, the field Number of Icons, it was informed the value 1000, so the number of stars besides, it determined by dividing the displayed value by 1000.

- **Display Value** : It will display the value and the chart at the same time, the value being on the left or right side of the bar or lines.
- **Number of Icons** : Determines the amount of icons (figures) that will appear for each value on this field. This value will be presented by the dividing the value of the column by the Number of Icons.
- **Chart width** : Width of the chart in pixels. This field is used by the option Bar Chart. In case it is not informed, the value used is 200 pixels.
- **Chart height** : Height of the chart in pixels. This field is used by the option Bar Chart.
- **Margin** : Charts margin. This option is only used by the bar chart.
- **Image / Positive Color** : The way that it is informed will variate depending on the options below:
 - **Bar Chart** : Inform a color for the field that will be used to form the chart image. Click the icon beside to choose the color.
 - **Line Chart** : Inform an image name for the field that will be used like an icon. Click the icon beside to choose the image "Icon".
- **Image / Negative Color** : The information passed to this field follows the same criteria as the Image / Positive Color field. This parameter will be used in the case the value is negative.

PDF Configuration

Line Break

This option allows breaking the line in the PDF file when exported.

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

 *Bar Chart Interface.*

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

 *Chart layout settings interface.*

1. [Home](#)
2. [Applications](#)
3. Grid
4. Currency

Currency

General Settings

Currency field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Currency, it is allowed to define the format of a currency.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Accumulated** : When enables, define that the field will be accumulated with the values of another field. This option is only available on virtual fields.
- **Accumulator field** : Only available when the option "Accumulated" is enabled. This configuration defines which field will be used to accumulate the values. The chosen fields needs to be Integer, Currency or Decimal.
- **Grid Mask** : Defines the mask for the field display. There are two typed of masks described below:

Character	Description
X	Placeholder to any character. Replaced by any character. If number of characters entered are less then the mask size, the field value is completed with zeros (Filling full size field entry is required).
Z	Replaced by any character retrieved from database. Suppress zeros at field left (Complete field filling is optional). When used combined with the mask character X it should be placed at the mask left.

Mask Examples

Mask	Field Value	Formatted Value
(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
(zz) xxxx - xxxx	1234567890	(12) 3456 - 7890
(zz) xxxx - xxxx	12345678	() 1234 - 5678
(zz) xxxx - xxxx	0012345678	() 1234 - 5678

- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the datatype of field in the database.

Values Format

Interface of Values Format.

- **Regional Settings** : Allows to apply the Regional Settings to format the number of the fields. When not enabled, you will view attributes grouped, Negative Symbol and negative number format.
- **Currency Format** : Defines the content of the field that if it will be presented with the currency format.
- **Currency Symbol** : Defines the character that represents the Currency Symbol.
- **Digit Grouping Separator** : Defines the character that will be used to separate the thousand.
- **Decimal Separator** : Defines the character that will be used to separate the decimal.
- **Negative sign** : Defines the character that will be used to display negative numbers.
- **Negative number format** : Defines the position of the Negative sign em relation to the value.
- **Color of Negative** : Defines a color for the field when the value is negative.
- **Decimal Precision** : Number of decimal places. This value is part of the field size.
- **Complete with Zeros** : Defines the decimal places that will or not be completed zeros to the right. Example: The value is 125,50, but the decimal precision is 3. If the option is enabled, then the value will 125,500.
- **Display value in full** : The value will be displayed in full. Example: 2018(Two thousand eighteen).

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field

and the value returned from the select command separated by the character in the "Separated by" field.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	

Attribute	Value	Lookup	Description
8		Reading	
16		Music	

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
- **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
- **Summary** : Applies the lookup only for the summary (HTML and PDF).
- **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

Line Chart

Using this guide, you can define ac chart for each line.

- **Line Chart Type** : There are two types of charts that can be used. Below you can see their description:
- **Bar Chart** : This option allows to display a bar chart relating to the column that you are working with. The size of the bars is done by math relating to the Grand Total of the column. The value informed in the width of the chart defines the size of the images that will be displayed.

In the example shown above, the width of the Chart is 200 pixels.

- **Line Chart** : This option allows to display a Line Chart relating to the column that you are working with . The amount of lines are done by dividing the value of a column by the value informed to the number of icons.

In the example above, the field Number of Icons, it was informed the value 1000, so the number of stars besides, it determined by dividing the displayed value by 1000.

- **Display Value** : It will display the value and the chart at the same time, the value being on the left or right side of the bar or lines.
- **Number of Icons** : Determines the amount of icons (figures) that will appear for each value on this field. This value will be presented by the dividing the value of the column by the Number of Icons.
- **Chart width** : Width of the chart in pixels. This field is used by the option Bar Chart. In case it is not informed, the value used is 200 pixels.
- **Chart height** : Height of the chart in pixels. This field is used by the option Bar Chart.
- **Margin** : Charts margin. This option is only used by the bar chart.
- **Image / Positive Color** : The way that it is informed will variate depending on the options below:
 - **Bar Chart** : Inform a color for the field that will be used to form the chart image. Click the icon beside to choose the color.
 - **Line Chart** : Inform an image name for the field that will be used like an icon. Click the icon beside to choose the image "Icon".
- **Image / Negative Color** : The information passed to this field follows the same criteria as the Image / Positive Color field. This parameter will be used in the case the value is negative.

PDF Configuration

Line Break

This option allows breaking the line in the PDF file when exported.

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

 *Bar Chart Interface.*

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

 *Chart layout settings interface.*

1. [Home](#)
2. [Applications](#)
3. Grid
4. Percent (Calculated)

Percent (Calculated)

General Settings

Percentage field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Percentage, it is allowed to define the format of a percentage.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Accumulated** : When enables, define that the field will be accumulated with the values of another field. This option is only available on virtual fields.
- **Accumulator field** : Only available when the option "Accumulated" is enabled. This configuration defines which field will be used to accumulate the values. The chosen fields needs to be Integer, Currency or Decimal.
- **Grid Mask** : Defines the mask for the field display. There are two typed of masks described below:

Character	Description
X	Placeholder to any character. Replaced by any character. If number of characters entered are less then the mask size, the field value is completed with zeros (Filling full size field entry is required).
Z	Replaced by any character retrieved from database. Suppress zeros at field left (Complete field filling is optional). When used combined with the mask character X it should be placed at the mask left.

Mask Examples

Mask	Field Value	Formatted Value
(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
(zz) xxxx - xxxx	1234567890	(12) 3456 - 7890
(zz) xxxx - xxxx	12345678	() 1234 - 5678
(zz) xxxx - xxxx	0012345678	() 1234 - 5678

- **Do Not Repeat Value** : Don't repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Applies the Regional Settings to format the number of the fields. When not enabled, you will view attributes grouped, Negative Symbol and negative number format.
- **Digit Grouping Separator** : Defines the character that will be used to separate the thousand.
- **Decimal Separator** : Defines the character that will be used to separate the decimal.
- **Negative sign** : Defines the character that will be used to display negative numbers.
- **Negative number format** : Defines the position of the Negative sign em relation to the value.
- **Color of Negative** : Defines a color for the field when the value is negative.
- **Decimal Precision** : Number of decimal places. This value is part of the field size.
- **Complete with Zeros** : Defines the decimal places that will or not be completed zeros to the right. Example: The value is 125,50, but the decimal precision is 3. If the option is enabled, then the value will 125,500.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

Attribute Value Lookup Description

1 (Sports) + 4 (Pleasure) + 8 (Reading) =13

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

Line Chart

Using this guide, you can define ac chart for each line.

- **Line Chart Type** : There are two types of charts that can be used. Below you can see their description:
- **Bar Chart** : This option allows to display a bar chart relating to the column that you are working with. The size of the bars is done by math relating to the Grand Total of the column. The value informed in the width of the chart defines the size of the images that will be displayed.

In the example shown above, the width of the Chart is 200 pixels.

- **Line Chart** : This option allows to display a Line Chart relating to the column that you are working with . The amount of lines are done by dividing the value of a column by the value informed to the number of icons.

In the example above, the field Number of Icons, it was informed the value 1000, so the number of stars besides, it determined by dividing the displayed value by 1000.

- **Display Value** : It will display the value and the chart at the same time, the value being on the left or right side of the bar or lines.
- **Number of Icons** : Determines the amount of icons (figures) that will appear for each value on this field. This value will be presented by the dividing the value of the column by the Number of Icons.
- **Chart width** : Width of the chart in pixels. This field is used by the option Bar Chart. In case it is not informed, the value used is 200 pixels.
- **Chart height** : Height of the chart in pixels. This field is used by the option Bar Chart.
- **Margin** : Charts margin. This option is only used by the bar chart.
- **Image / Positive Color** : The way that it is informed will variate depending on the options below:
 - **Bar Chart** : Inform a color for the field that will be used to form the chart image. Click the icon beside to choose the color.
 - **Line Chart** : Inform an image name for the field that will be used like an icon. Click the icon beside to choose the image "Icon".
- **Image / Negative Color** : The information passed to this field follows the same criteria as the Image / Positive Color field. This parameter will be used in the case the value is negative.

PDF Configuration

Line Break

This option allows breaking the line in the PDF file when exported.

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

 *Bar Chart Interface.*

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

 *Chart layout settings interface.*

- 1. [Home](#)
- 2. [Applications](#)
- 3. [Grid](#)
- 4. [HTML Image](#)

HTML Image


General Settings

 HTML Image field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as HTML Image, it allows to place an image to the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Image** : Defines an image that will be displayed. The icon "Select Image", lists all the standard images from scriptcase and also the image that you have uploaded to scriptcase. The icon "Upload an image" allows the developer to send an image to the server which is from another machine.
- **Border** : Border size for the image in Pixels.
- **Width** : Define the width of the image in Pixels.
- **Height** : Define the height of the image in Pixels.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

 Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Credit Card Number

Credit Card Number

General Settings

Credit Card Number Configuration Interface.

- **Data Type** : Select the type of field for the application. When it is defined as a Credit Card Number, you can define some rules for the display format of the Credit Card.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name would be "Client Name".
- **Do Not Repeat Value** : Do not repeat the value of the field in the case it is the same as the previous record.
- **SQL Type** : Informs the data type of the field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

PDF Configuration

Line Break

This option allows breaking the line in the PDF file when exported.

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

Chart layout settings interface.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. E-mail

E-mail

General Settings

Email field Configuration Interface.

- **Data Type** : Defines the type of field for the application. When it is defined as an Email, when you click on the field you be offered a choice for your email client and send an email to that specific email.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

PDF Configuration

Line Break

This option allows breaking the line in the PDF file when exported.

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

Chart layout settings interface.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. URL

URL

General Settings

URL field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a URL, it sets the value as a clickable link (Only if it contains a valid path).
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

PDF Configuration

Line Break

This option allows breaking the line in the PDF file when exported.

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.


Chart Layout

Chart layout settings interface.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. YouTube

YouTube


General Settings

 YouTube field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a YouTube, it allows to display a video from youtube on the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Display Mode** : Defines how the you video will display in the application.
- **Width** : Width of the video in pixels.
- **Height** : Height of the video in pixels.
- **Link Type** : The display settings of the link, if it is going to be a Text or a Button.
- **Link text** : A text that will link to the video.
- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

 Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Google Maps

Google Maps

General Settings

Google Maps field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as Google Maps, it will use the Google Maps API to display the map in the Grid Applications
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Home** : Defines what type of parameters will be used in the API.

Home Configuration Interface.

- **Display Mode** : Indicates the display mode of the map. It can be opened in a Modal or in a new Window.
- **Width** : Defines the width of the map that is going to be displayed.
- **Height** : Defines the height of the map that is going to be displayed.
- **Zoom** : Defines the initial Zoom (available from the Google API) of the Map location.
- **API Key** : API Key for authorization to use Google Maps in the Application. (Required only for the versions 2 or earlier of the Google API.)

The API Key is a unique key, composed by a string(text) alphanumeric, which is the license to use the service. When you subscribe to use the service, the key is tied to the domain and the directory of the server. All the pages that use the API needs to be in the same directory that was used for the subscription. In case you have a web server on your local machine, you just need to possess a key for testing, and to do this you only need to set http://localhost in the domain of the subscription.

To get your API Key access the site by [clicking here](#)

- **Link Type** : Defines how the link will be displayed.
- **Text Link** : Text to call the Map.
- **Marker Description** : Displays the description for each marker displayed on the map.
- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Date

Date

General Settings

Date field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Date and Time, you can define the format of the date.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

Format of Values without Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator attribute (Check the Regional Settings).
- **Display** : Offers a series of formats predefined for displaying dates.
- **Detail Mask** : Define the format of the field in the Grid, following the standards the PHP function Date.

```
d-m-Y      25-09-2009
F/Y       September/2009
j/n/Y \a\s g:i:s A 25/9/2001 as 14:30:11 PM
l, d \d\e F \d\e Y      Thursday, 25 of January of 2009
h:i:s     11:33:20
#h:i:s    123:43:27 (accumulating the
           hours)
```

- **Format Table**

- **Internal Format** : Allows to define the format the field when the SQL type is different from **Date**. You need to use the characters **Y**, **M** and **D** that correspond to **Year**, **Month** and **Day**.
 - **Example 1** : Date stored in a SQL field of the type **char(8)** where the positions **1** to **4** represents the year, the positions **5** and **6** is the month and the position **7** and **8** is the day, define the internal format like: **YYYYMMDD**
 - **Example 2** : Date stored in a SQL field of the type **char(6)** where the positions **1** and **2** represents the month and the positions **3** to **6** is the year, define the internal format like: **MMYYYY**

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

PDF Configuration

Line Break

This option allows breaking the line in the PDF file when exported.

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

Chart layout settings interface.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Time

Time

General Settings

Time field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as Time, it is possible to define a Time Format.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

Format of Values without Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator attribute (Check the Regional Settings).
- **Display** : Offers a series of formats predefined for displaying dates.
- **Detail Mask** : Define the format of the field in the Grid, following the standards the PHP function Date.

```
d-m-Y      25-09-2009
F/Y       September/2009
j/n/Y \a\s g:i:s A 25/9/2001 as 14:30:11 PM
l, d \d\e F \d\e Y Thursday, 25 of January of 2009
h:i:s     11:33:20
#h:i:s    123:43:27 (accumulating the
          hours)
```

- **Format Table**

- **Internal Format** : Allows to define the format the field when the SQL type is different from **Date**. You need to use the characters **A**, **M** and **D** that correspond to **Year**, **Month** and **Day**.
 - **Example 1** : Date stored in a SQL field of the type **char(8)** where the positions **1** to **4** represents the year, the positions **5** and **6** is the month and the position **7** and **8** is the day, define the internal format like: **YYYYMMDD**
 - **Example 2** : Date stored in a SQL field of the type **char(6)** where the positions **1** and **2** represents the month and the positions **3** to **6** is the year, define the internal format like: **MMYYYY**

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

PDF Configuration

Line Break

This option allows breaking the line in the PDF file when exported.

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

Chart layout settings interface.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Date and time

Date and time

General Settings

Date and Time field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Time, you can define the format of the time.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

Format of Values without Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator attribute (Check the Regional Settings).
- **Display** : Offers a series of formats predefined for displaying dates.
- **Detail Mask** : Define the format of the field in the Grid, following the standards the PHP function Date.

```
d-m-Y      25-09-2009
F/Y       September/2009
j/n/Y \a\s g:i:s A 25/9/2001 as 14:30:11 PM
l, d \d\e F \d\e Y      Thursday, 25 of January of 2009
h:i:s     11:33:20
#h:i:s    123:43:27 (accumulating the
           hours)
```

- **Format Table**

- **Internal Format** : Allows to define the format the field when the SQL type is different from **Date**. You need to use the characters **Y**, **M** and **D** that correspond to **Year**, **Month** and **Day**.
 - **Example 1** : Date stored in a SQL field of the type **char(8)** where the positions **1** to **4** represents the year, the positions **5** and **6** is the month and the position **7** and **8** is the day, define the internal format like: **YYYYMMDD**
 - **Example 2** : Date stored in a SQL field of the type **char(6)** where the positions **1** and **2** represents the month and the positions **3** to **6** is the year, define the internal format like: **MMYYYY**

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

PDF Configuration

Line Break

This option allows breaking the line in the PDF file when exported.

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.


Chart Layout

Chart layout settings interface.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Image (Database)

Image (Database)


General Settings

 Upload field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as an Image (Database), all the Images files are stored and loaded directly from the Database.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Image Border** : Width of the Image border. Value in Pixels.
- **Image Height** : Image height size. Value in Pixels.
- **Image Width** : Image width size. Value in Pixels.
- **Maintain Aspect** : Maintains the original aspect ratio of the image when resizing it.
- **Open in Another Window** : Allows to open the image in another window.
- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

 Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

1. [Home](#)
2. [Applications](#)
3. Grid
4. Image (File Name)

Image (File Name)

General Settings

Upload field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as an Image (File Name), all the images files are stored and loaded in a directory of the server (Only the image name is stored in the database).
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Image Border** : Width of the Image border. Value in Pixels.
- **Image Height** : Image height size. Value in Pixels.
- **Image Width** : Image width size. Value in Pixels.
- **Maintain Aspect** : Maintains the original aspect ratio of the image when resizing it.
- **Open in Another Window** : Allows to open the image in another window.
- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.


Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Document (Database)

Document (Database)


General Settings

 Upload field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Document (Database), all the document files are stored and loaded directly from the Database.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Icon** : Displays an icon beside the field to identify the type of document.
- **File Name** : Defines the field that contains the name of the document stored in the database (the field that contains this information also needs to be stored in the database).
- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

 Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

1. [Home](#)
2. [Applications](#)
3. Grid
4. Document (File Name)

Document (File Name)

General Settings

Upload field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Document (File Name), all the document files are stored and loaded in a directory of the server (Only the document name is stored in the database).
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Sub-folder** : Sub-folder name that the files are stored.
- **Icon** : Displays an icon beside the field to identify the type of document.
- **File Name** : Defines the field that contains the name of the document stored in the server (the field that contains this information also needs to be stored in the database).
- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Bar code

Bar code

General Settings

 Configuration Interface of the Barcode Field.

- **Data Type** : DataType of the field for the application.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name would be "Client Name".
- **SQL Type** : Database field type.

Watch below a video showing an example about the Barcode field:



Values Format

 Configuration Interface of the Barcode Field.

- **Type** : Type of Barcode.
- **Text** : Barcode Text for illustration purposes.
- There are **18 types of barcodes** that are listed below:

 Barcode configuration interface.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.


 Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

1. [Home](#)
2. [Applications](#)
3. Grid
4. QRCode

QRCode

General Settings

 QRCODE field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a QRCODE, allows you to set values into a QRCODE.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **SQL Type** : Informs the data type of field in the database.


Watch below a video showing an example with QRCODE



Values Format


 Interface of Values Format.

- **Level of error correction**: Codewords are 8 bits long and use the Reed-Solomon error correction algorithm with four error correction levels. The higher the error correction level, the less storage capacity.
- **Image Size** : Size of the QRCODE.
- **Margin** : Margin of the QRCODE.

 Interface of Values Format.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.


 Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Signature

Signature

General Settings


 Configuration Interface of the Signature Field.

The signature field will help you creating more sophisticated forms and making it possible to store signatures in your database. Inside our development environment we have specific settings that will help you to customize your field, those options are:

- **Data Type** : You can define the type of field for the application. When it is defined as a text, it accepts letters, numbers and special characters.
- **Label** : Lets you define a label to the field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Background color** : Defines a color to the field background by using a valid hexadecimal color value or by choosing a color from the color palette.
- **Pen color**: Set a color to the pen by using a valid hexadecimal color value or by choosing a color from the color palette.
- **Width** : Lets you define a width to the field.
- **Height** : Set a height to the field.
- **Subtitle** : Defines the subtitle that will be displayed beside the field.
- **Initial Value** : Lets you define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
- **Save Variable** : Save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Set the name for the session variable that will receive the field value.
- **Disabled Field** : Define if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : Displays the HTML Object that will be used to display the field in the form.
- **SQL Type** : Displays the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of ScriptCase, there are the same attributes available in Interface.

 Display Settings configuration Interface.

- **CSS of the Title**
 - **Font** : Select the font type, that will be applied to the application field title.
 - **Font Size** : Defines the font size, that will be applied to the application field title.
 - **Font Color** : Choose a color to the font by using a valid hexadecimal color value or from the color picker.
 - **Background Color** : You can define the color for the field by using a valid hexadecimal color value or from the color picker.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Set the underline style to the font.
 - **Border style** : Choose a style for the border.
 - **Collapse** : Defines the collapse for the border.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Choose a color for the border, using a color palette to apply to the title.
 - **Horizontal Alignment** : Position the label of the field in the wanted position (left,right,center and justify).
 - **Vertical Alignment** : Position the label of the field in the wanted position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : Defines a width for the title of the field.
 - **Height** : Set a height for the title of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Rating

Rating

This field allows the developer to create a rate field to display the data previously added using stars (or any other image).

Description of the attributes available for configuration.

Data type

Define the type of the field in the application. In this case we must select the type **Rating**.

Label

Define the application title that will be showed when the application is executed. The terminology used in the interface of your application is very important for the usability, and we need to use familiar terms for the final user, instead of use terms from the system.

Example, this field "rating" that is named **cmp_rating** however, the customer would have a better understanding if the label uses **Product Rating**.

Besides use a fixed text, the **Label** attribute allows the use of **Langs** to define the field title, allowing the [internationalization of your application](#).

Subtitle

Define the subtitle that will be shown in the field below the ratings. **Example**: "Thank you for your feedback!".

As in the **Label**, the **subtitle** attribute also allows the use of **Langs** for [internationalization of your application](#).

Amount of icons

Define the number of icons that will be displayed in the field. The value set in this attribute must be according to the way the evaluation will be realized.

To evaluate using Stars normally it is used five stars, so , we must to inform 5 in this attribute. After we set the images that will be displayed in the attribute **Enabled icon** and **Disabled icon** the result will be the same as in the example:

To use Like/Dislike **for example** we must inform 1 in the **amount of icons**, so only one image will be displayed and will change according to the click. In this case, the storage in the database will be 1 for like (enabled icon) and 0 for dislike (disabled icon). **You can see the field configured below as like/dislike:**

Enabled icon

Define the icon that will be shown for the amount selected. Following the like/dislike example, in this option, we must insert the Like image . We must do the same for a star evaluation, where for default we must insert the stars filled .

Disabled icon

Define the icon that will be shown for the not selected icons. In this attribute, we set an image to represent the opposite of the previous images set in the **Enabled icon**.

Still referring the example of the Like/Dislike, in this attribute we must set the Dislike .

As in the stars example, we must insert the image with the empty stars .

SQL Type

Inform the type of the field in the database.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

• CSS of the Title

- **Font** : Allows to choose the font type, that will be applied to the application field title.
- **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
- **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.

• CSS of the Field

- **Font** : Allows to choose the font type, that will be applied to the application field.
- **Font Size** : Allows to choose the the font size, that will be applied to the application field.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
- **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).

- **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the field.
- **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

Blocks

Blocks are “containers” where you can position the application fields of Forms, Controls, or Slide Grids.

Scriptcase creates applications with one block by default. You can add more blocks as you wish, to organize it in the best way.

See below, the Columns Organization, and where you can define the position of the next block: beside or below the current one.

		Block	Title	Label		Fields		Organization				
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse		
Pag1												
			form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Pag2												
			Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Blocks not shown												

Application Block configuration

On the left side of each block, there are two icons, the first one to edit the information of the block and the second one to delete the block.

Organizing The Position Of The Blocks

See below how to modify the display order of the Blocks in one Page.

Click and drag the block that you desire to modify to its new position.

		Block	Title	Label		Fields		Organization				
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse		
Pag1												
			form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Pag2												
			Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Blocks not shown												

Application Block Display configuration

- see how to remove a block from the display

Click on the block desired and drag it to the item “Blocks not Shown”. This way, you can also drag the block to another page if desired. See the images below.

		Block	Title	Label		Fields		Organization				
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse		
Pag1												
			form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Pag2												
			Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Blocks not shown												
			Block 3	Block 3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼

Application Block Display configuration

		Block	Title	Label		Fields		Organization				
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse		
Pag1												
			form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Pag2												
			Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Blocks not shown												
			Block 3	Block 3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼

Application Block Display configuration

Block

- **ame:** The name that identifier the Block.
- **abel:** Title of the block to display in the application.

Title

- **isplay:** It controls the display of the block title.

Label

- **isplay:** It controls the display of the field labels of the block.
- **osition:** Options to display label :
 1. **bove:** Display the label above the field.
 2. **eside:** Display the label beside the field.
 3. **elow:** Display the label below the field.

Fields

- **olumns:** Amount of columns side by side in the block.
- **osition:** The way to display the fields in the block:
 1. **elow:** Display the fields one below the other respecting the number of columns.
 2. **eside:** Display the fields one beside the other respecting the number of columns.
 3. **ine:** Display the fields one beside the other with no tabulation.

Organization

- **ext:** The way to display the blocks in the page:
 1. **elow:** Set to show the following block below the current one.
 2. **eside:** Set to show the following block beside the current one.
 3. **abs:** Set to show the following block in a different tab then the current one.
- **idth:** Set the block width in pixels or percentages. Use the symbol "%" to indicates the value in percentage.
- **ollapse:** Enables the option to close the block.

Create A New Block

To include new blocks in an Application, click on the button [Create New Block](#). Then, enter the name and label of the block in the following interface and finish by click on Create.

Creating application blocks configuration


Name






Name of the Block.

Label

Title of the block to display in the application.

Edit Blocks

To edit a block, click on the icon , that is on the left side of the block. Then you can see the following interface to define the parameters of the blocks. Click on Save to finish.

EDIT BLOCKS	
ATTRIBUTE	VALUE
Name	<input type="text" value="form_orders"/>
Title	<input type="text" value="form_orders"/>
Display Title	<input type="radio"/> Yes <input checked="" type="radio"/> No
Title Font	<input type="text" value=""/> Aa
Font Size	<input type="text" value=""/>
Font Color	<input type="text" value=""/> 
Background Color	<input type="text" value=""/> 
Background image	<input type="text" value=""/> 
Title Height	<input type="text" value="20"/> pixels
Horizontal Alignment	<input type="text" value=""/>
Vertical Alignment	<input type="text" value=""/>
Display Label	<input checked="" type="radio"/> Yes <input type="radio"/> No
Columns	<input type="text" value="1"/>
Columns Width	<input type="text" value="Calculated"/>
Label Color	<input type="text" value=""/> 
Fields Organization	<input type="text" value="Beside"/>
Label Position	<input type="text" value="Beside"/>
Next Block	<input type="text" value="Below"/>
Border Color	<input type="text" value=""/> 
Border Width	<input type="text" value="0"/> pixels
Block Width	<input type="text" value="100%"/>
Block Height	<input type="text" value=""/>
Cell Spacing	<input type="text" value=""/> pixels
Collapse	<input type="text" value="Start open"/>

Application Block editing interface

Name

Name of the block. ##### Title
 Block title for display. ##### Display Title
 This option, when active, allows displaying the block title. ##### Title Font
 Set the font family of the block title. ##### Font Size
 Set the font size of the block title. ##### Font Color
 Set the font color of the block title. ##### Background Color
 Set the Background Color of the block title. ##### Background image
 Set a Background image for the block title. ##### Title Height
 Height in pixels of the block title line. ##### Horizontal Alignment
 Horizontal Alignment of the block title (Left, Center, and Right). ##### Vertical Alignment
 Vertical Alignment of the block title (Top, Middle, and Bottom). ##### Display Label
 Display the labels of the fields in the block. ##### Columns
 Amount of field columns in a block. ##### Columns Width
 Set the field column width of the block. ##### Label Color
 Color of the field labels. ##### Fields Organization
 The way to display the fields in the block. ##### Label Position
 Set the position of the field labels of the block. ##### Next Block
 Set the position of the following block relating to the current one. ##### Border Color
 The border Color for the block. ##### Border Width
 The border Width for the block. ##### Block Width
 The width for the block. ##### Block Height
 The Height for the block. ##### Cell Spacing
 The Cell Spacing in the block. ##### Collapse
 It enables the option to close the block.

This module lets you editing display options for the application. You can apply visualization themes, organize blocks, define values, and the display format of the Header and Footer.

Layout Settings

ScriptCase creates an app by using default values defined per project, but you can change those values for a specific application on this interface.

▾ LAYOUT SETTINGS

Header Template	<input type="text" value="Flat"/>	Template name used for the application header.
Footer Template	<input type="text" value="Default"/>	Template name used for the application footer.
Button	<input type="text"/>	Use different buttons than what was defined in the color scheme.
Themes	<input type="text" value="Sc9_Rhino"/>	Use different themes from the one defined for the color scheme

Header

|< << >> >| xxyyzz xxxxx yyyyy ▾

Block 1

Name

Type Male Female

Address*

Groups* Male Female

Countries

Address

Drag & Drop files here

Photos

Image1.png ✓

Image2.png ✗

Captcha

Application Layout Settings

Header Template

Allows choosing the look of the application Header.

Footer Template

Allows choosing the look of the application Footer.

Button

Allows choosing the button theme for the application.

Themes

Choose one of the selected themes in the project properties. It defines the look of the application, like colors, fonts, and others.

Header & Footer

Header

Here we can define if we want to display the header or not, the application title, and fill the header variables. The header looks according to the selected template in the Layout settings interface.

HEADER	
ATTRIBUTE	VALUE
Display Header	<input checked="" type="checkbox"/>
Insertion Title	{lang_othr_frmi_titl} - orders
Update Title	{lang_othr_frmu_titl} - orders
Header Variables	
SC_HEADER	Title
SC_VALUE	Date

Application Header Configuration

Display Header


This option determines if we want to display the header or not.

Title

Allows informing the title of the application.

Header Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **itle**: It displays the value of "**Application Title**" in the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Depending on the Application, you may have more than one title option.

Footer

FOOTER	
ATTRIBUTE	VALUE
Display Footer	<input type="checkbox"/>
Footer Variables	
NM_FOOTER_DATA	
NM_FOOTER_LOGO	
NM_FOOTER_TITLE	

Application Footer Settings


This page may change depending on the footer template chosen in the Layout Settings.

Display Footer

This option determines if we want to display the footer or not.

Footer Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Scriptcase has Events to allow the developers to customize the application code. By using the events, the developers can program custom actions at a specific execution time (e.g. After a record is inserted, when it is loaded, upon submitting a Form, ...) and for a specific application type. In the events area you can use global and local variables, JavaScript, CSS, HTML, PHP codes and also Scriptcase macros.

onApplicationInit

This event occurs only once, when the application is loaded and before the application executes the SQL. It can be used for data processing or checking variables.

Example: `if ([glo_var_dept] != 'financial'){ sc_redir(app_x.php); }`

onNavigate

This event occurs when navigating between the application pages.

OnScriptInit

This event occurs when the application scripts are started. It runs before the OnLoad events(when present), however, it will always be executed when running, reloading or navigating the application.

onRecord

This event is executed at the moment the record line is loading. The OnRecord event is a loop that loads the records by each row.

onHeader

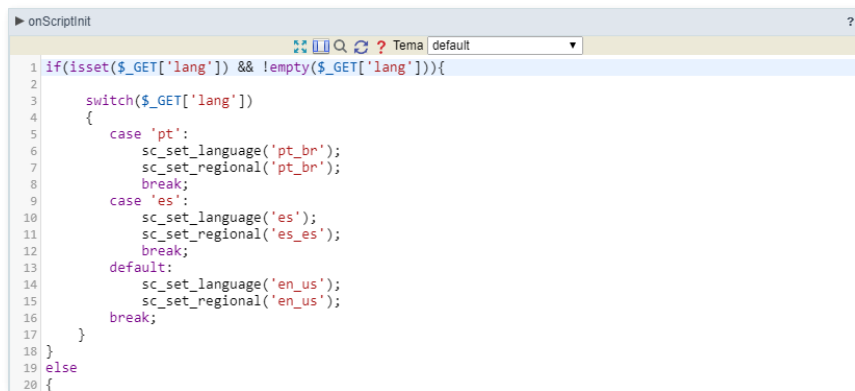
This event runs when the header of the application is loaded.

onFooter

This event runs when the footer of the application is loaded.

Code Editor

Scriptcase code editor is used within events and has a series of shortcuts to help in the applications development.







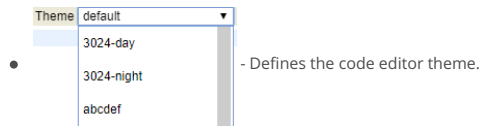
```

onScriptInit
1 if(isset($_GET['lang']) && !empty($_GET['lang'])){
2
3     switch($_GET['lang'])
4     {
5         case 'pt':
6             sc_set_language('pt_br');
7             sc_set_regional('pt_br');
8             break;
9         case 'es':
10            sc_set_language('es');
11            sc_set_regional('es_es');
12            break;
13        default:
14            sc_set_language('en_us');
15            sc_set_regional('en_us');
16            break;
17    }
18 }
19 else
20 {

```

At the top section of the editor, there are some options that allows us to change the editor theme or expand it, for example. These options can also be activated using the hotkeys.

-  - Increases the area occupied by your code editor.
-  - Expand the sides of the code editor.
-  - Activates the search in the code editor.
-  - Enables replace in the code editor.



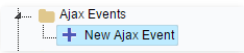
The hotkeys are available in all browsers compatible with Scriptcase.

[Click Here](#) to view the *Scriptcase hotkeys documentation*.

OnClick

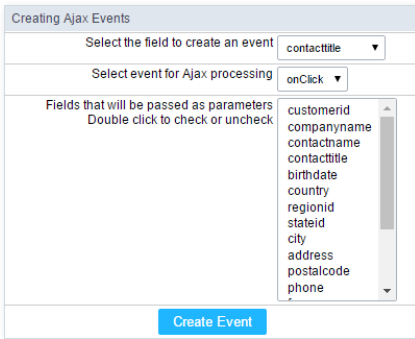
The ajax event OnClick is executed when the field that it's based on is clicked.

- creating a new ajax event



Creating a new ajax event

- selecting a field



Creating Ajax Events

Select the field to create an event contacttitle

Select event for Ajax processing onClick

Fields that will be passed as parameters
Double click to check or uncheck

- customerid
- companyname
- contactname
- contacttitle
- birthdate
- country
- regionid
- stateid
- city
- address
- postalcode
- phone

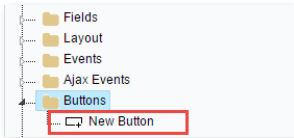
Create Event

Selecting a field of the ajax event

Choose a field to create an event To define in which field the event will be added to.

Choose an event that runs the ajax Defines which event will be added to the field.

In addition to the buttons that comes automatically with the applications, you can also create your own buttons. All the buttons are displayed within the application toolbar.

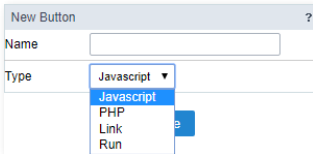


New buttons creation settings

Creating a new button

To create a new button, click on the “New Button” option and enter a name and a button type.

The button Types are: JavaScript, PHP, Link and Run.



Grid buttons type

Deleting a button

To delete a button click on Delete icon (represented by a recycle bin image) in the right of the button name, at the application menu under the Buttons option.



Deleting a button

JavaScript

Display Mode

You can configure the display mode of the javascript button in Image, Button or Link.

Button

ATTRIBUTE	VALUE
Display Mode	Button
Label	JavaScript
Hint	
Confirmation Message	
Type	JavaScript

Setting up Javascript Button.

Display Mode You can select the display mode for the javascript button in this option.

Label Text of the button that will be display in the application while executing.

Hint Hint message for the button. (Displayed when the mouse hovers the button).

Confirmation Message Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.

Type Description of the created button.

Image

ATTRIBUTE	VALUE
Display Mode	Image
Icon	
Hint	
Confirmation Message	
Type	JavaScript

Setting up Javascript Button.

Display Mode You can select the display mode for the javascript button in this option.

Icon Allows you to inform the icon that will be displayed on the button while the execution of the application.

Hint Hint message for the button. (Displayed when the mouse hovers the button).

Confirmation Message Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.

Type Description of the created button.

Link

ATTRIBUTE	VALUE
Display Mode	Link
Label	JavaScript
Hint	
Confirmation Message	
CSS Style	default
Type	JavaScript

Setting up Javascript Button.

Display Mode You can select the display mode for the ajax button in this option.

Label Text of the button that will be display in the application while executing.

Hint Hint message for the button. (Displayed when the mouse hovers the button).

Confirmation Message Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.

CSS Style Name of the CSS class, style created in the layout editor.

Type Description of the created button.

Code Block

Javascript button coding block.

In this block, only JavaScript is accepted.

PHP

Display Mode

You can configure the display mode of the PHP button in Image, Button or Link.

Button

ATTRIBUTE	VALUE
Display Mode	Button
Label	PHP
Hint	
Confirmation Message	
Reload quantity of records	No
Type	PHP
Target	Same Window

Setting up PHP Button.

Display Mode You can select the display mode for the link button in this option.

Label Text of the button that will be display in the application while executing.

Hint Hint message for the button. (Displayed when the mouse hovers the button).

Confirmation Message Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.

Reload quantity of records This option is used to update the amount of records in the application.

Type Description of the created button.

Target Defines the window destination where the code will run (Open in the same Window, Open in another Window, Modal).

Image

ATTRIBUTE	VALUE
Display Mode	Image
Icon	
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
Reload quantity of records	No
Type	PHP
Target	Same Window

Setting up PHP Button.

- Display Mode** You can select the display mode for the link button in this option.
- Icon** Allows you to inform the icon that will be displayed on the button while the execution of the application.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Reload quantity of records** This option is used to update the amount of records in the application.
- Type** Description of the created button.
- Target** Defines the window destination where the code will run (Open in the same Window, Open in another Window, Modal).

Link

ATTRIBUTE	VALUE
Display Mode	Link
Label	PHP
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
CSS Style	default
Reload quantity of records	No
Type	PHP
Target	Same Window

Setting up PHP Button.

- Display Mode** You can select the display mode for the link button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- CSS Style** Name of the CSS class, style created in the layout editor.
- Reload quantity of records** This option is used to update the amount of records in the application.
- Type** Description of the created button.
- Target** Defines the window destination where the code will run (Open in the same Window, Open in another Window, Modal).

Code Block

```

1 *Ajax button coding block. *

```

In this block, you can use macros, PHP code and JavaScript.

Link Button

Display Mode
You can configure the display mode of the link button in Image, Button or Link.

Button

ATTRIBUTE	VALUE
Display Mode	Button
Label	Link
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
Type	Link


Link

Setting up Link Button.

Display Mode	You can select the display mode for the link button in this option.
Label	Text of the button that will be display in the application while executing.
Hint	Hint message for the button. (Displayed when the mouse hovers the button).
Confirmation Message	Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
Type	Description of the created button.

Image

► Button Settings: Link

ATTRIBUTE	VALUE
Display Mode	Image ▾
Icon	<input type="text"/> 
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
Type	Link

[Link](#)

Setting up Link Button.

Display Mode	You can select the display mode for the link button in this option.
Icon	Allows you to inform the icon that will be displayed on the button while the execution of the application.
Hint	Hint message for the button. (Displayed when the mouse hovers the button).
Confirmation Message	Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
Type	Description of the created button.

Link

► Button Settings: Link

ATTRIBUTE	VALUE
Display Mode	Link ▾
Label	Link <input type="text"/>
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
CSS Style	default ▾
Type	Link

[Link](#)

Setting up link Button.

Display Mode	You can select the display mode for the ajax button in this option.
Label	Text of the button that will be display in the application while executing.
Hint	Hint message for the button. (Displayed when the mouse hovers the button).
Confirmation Message	Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
CSS Style	Name of the CSS class, style created in the layout editor.
Type	Description of the created button.

Setting Up The Link

- electing the applications

Link between applications - Application List

Select the application that will be called:

Application All By folder By type

- calendar_events
- chart_customers
- dashboard
- form_customers
- form_employees
- form_orders
- form_sec_users
- grid_categories

[← Back](#) [Next →](#) [Help](#)

Choosing the application for the button link.

You should select an application to be called from the button link.

- ink Parameters

Choosing the parameters for the button link.

Field Allows you to use an existing field from the current application as a parameter for the link.

Variable Allows you to use a global variable from the current application as a parameter for the link.

Fixed Allows you to inform a fixed value as a parameter for the link.

Empty No value will be passed as a parameter for the link.

- ink Properties (Grid)

Configuring the properties for the link button when the destined application is a Grid.

Link Operation Mode How the link will open (Open in the same Window, Open in another Window, Modal).

Exit URL for the target application URL or an application that be redirect to when exiting the Grid application.

Initial Mode Allows you to define the initial mode of the grid application (Search or Grid).

Number of Lines Allows you to define the amount of lines displayed in the Grid.

Number of Columns Allows you to define the amount of columns displayed in the Grid.

Paging Enable the paging in the Grid.

Display Header Enable the Grid Header.

Active Navigation Buttons Enable the navigation button (First, Back, Next and Last) in the Grid.

- ink Properties (Form)

Configuring the properties for the link button when the destined application is a Form.

Link Operation Mode How the link will open (Open in the same Window, Open in another Window, Modal).

Exit URL for the target application URL or an application that be redirect to when exiting the Form application.

Enable insert button on target application Enable the "New" button in the Form Application.

Enable update button on target application Enable the "Update" button in the Form Application.

Enable delete button on target application Enable the "Delete" button in the Form Application.

Enable navigation button on target application Enable the navigation button (First, Back, Next and Last) in the Form.

Enable button to edit a grid record Enable the buttons that allow you to edit the records of a Grid

RUN

Display options

We can configure "Run" button display as Link, Image or Button:


Button

ATTRIBUTE	VALUE
Display Mode	Button
Label	Run
Hint	
Confirmation Message	
Reload quantity of records	No
Type	Run
Target	Same Window

Grid's Run Button settings - Button type

- Display Mode** Combo box to select the button display option, you can choose button, image or link.
- Label** Application button title (text that will be displayed for the button within the application)
- Hint** Using this option you can set a message for the button hint
- Confirmation Message** Using this option you can set a confirmation message that will be displayed when the button is pressed. Leave it empty if you don't need to display any message.
- CSS Style** CSS for the the button, if you do not change this option Scriptcase will apply the default application/project CSS. You can customize the buttons' CSS using the option within the main menu "Layout » CSS Buttons"
- Reload quantity of records** Option used to update the amount of application records.
- Type** Button type description.
- Target** This option allows you to set the target window where you will run the button code (the same window, other window, modal)

Image

ATTRIBUTE	VALUE
Display Mode	Image
Icon	
Hint	
Confirmation Message	
Reload quantity of records	No
Type	Run
Target	Same Window

Grid's Run Button settings - Image type

- Display Mode** Combo box to select the button display option, you can choose image, button or link.
- Icon** Button display icon. You can use this option to select an image (from Scriptcase images or from you computer) to represents the button within the application toolbar.
- Hint** Using this option you can set a message for the button hint
- Confirmation Message** Using this option you can set a confirmation message that will be displayed when the button is pressed. Leave it empty if you don't need to display any message.
- Reload quantity of records** Option used to update the amount of application records.
- Tipo** Button type description.
- Target** This option allows you to set the target window where you will run the button code (the same window, other window, modal)

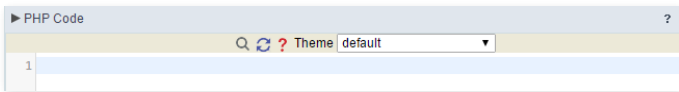
Link

ATTRIBUTE	VALUE
Display Mode	Link
Label	Run
Hint	
Confirmation Message	
CSS Style	default
Reload quantity of records	No
Type	Run
Target	Same Window

Grid's Run Button settings - Link type

- Display Mode** Combo box to select the button display option, you can choose link, button or image.
- Label** Text that will be displayed on the button (on running application).
- Hint** Application button title (text that will be displayed for the button within the application)
- Confirmation Message** By using this option you can set a confirmation message that will be displayed when the button is pressed. Leave it empty if you don't need to display any message.
- CSS Style** CSS class name, style created in the theme buttons editor .
- Reload quantity of records** Option used to update the amount of application records.
- Type** Button type description.
- Target** This option allows you to set the target window where you will run the button code (the same window, other window, modal)

Coding Area




Grid's Run Button settings - Coding area

- here are two types of events in the Run button
 - **nRecord** : Runs after processing on each record selected.
 - **nFinish** : Runs after processing all records selected.

In this coding area you can use Scriptcase macros, PHP and JavaScript.

imagem_ligacao_geral_consulta

The Grid Detail Module allows you to organize the information displayed according to its significance. It's a complementary application that displays the data in a detailed Form. In order to see the Grid Detail within the Grid the final user needs to click on  that is displayed for each record.

To enable the Grid Detail Module, access the "Grid Modules" and enable the check box, it uses to come enabled by default:

Detail Settings

In this section, the user can define how the detail of a particular record should be displayed when it's selected and which registry information should be displayed in this detail.

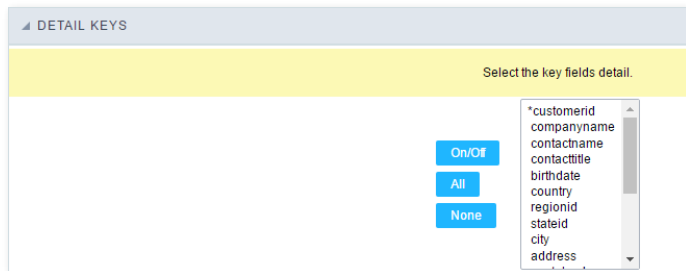


ATTRIBUTE	VALUE	DESCRIPTION
Select the fields to display in detail page.		

- **tributes**
 - **ields selection for the detail page.** : Allows you to select the desired fields. Just click the field and then click the "On/Off" button to enable/disable the fields.

Detail Keys

Through this interface you can chose the fields that will be part of the "WHERE" clause, it will return the corrected information according to the selected record. ScriptCase uses to identifies the table primary keys, that is being used within the application SQL, automatically. However, sometimes it is necessary to set the primary key manually, especially in cases where the application has a SQL JOIN, involving more than one table.



ATTRIBUTE	VALUE	DESCRIPTION
Select the key fields detail.		

Select here the fields that are keys to the Detail. The application will execute another select command, and you can configure which fields will be passed for this "WHERE" clause:

- **n/Off** : Select the field to be displayed in detail. The selected field (On) gets an asterisk.
- **ll** : Check "On" to all fields.
- **one** : Check "Off" to all fields.

Desktop Toolbar

The application toolbar is divided in two parts: superior and inferior. It is possible to select the buttons for each part independently. A button can even be part of both parts simultaneously.

ATTRIBUTE	VALUE	DESCRIPTION
Configure the toolbar below for a "classic web version " and also for a "mobile version".		
<div style="display: flex; justify-content: space-between;"> Desktop Mobile </div> <div style="display: flex; justify-content: space-around;"> <div style="width: 45%;"> <p>Top Toolbar</p> <ul style="list-style-type: none"> Export PDF Print Navigation Exit Separator </div> <div style="width: 10%; text-align: center;"> <ul style="list-style-type: none"> ⏪ ⏩ ⏴ ⏵ ⏴ ⏵ </div> <div style="width: 15%;"> <p>Left</p> <ul style="list-style-type: none"> Center PDF Print Exit Right </div> <div style="width: 10%; text-align: center;"> <ul style="list-style-type: none"> ⏴ ⏵ ⏴ ⏵ ⏴ ⏵ </div> <div style="width: 15%;"> <p>Group</p> <p>Add Edit Delete</p> </div> </div> <hr/> <div style="display: flex; justify-content: space-around;"> <div style="width: 45%;"> <p>Bottom Toolbar</p> <ul style="list-style-type: none"> Export PDF Print Navigation Exit Separator </div> <div style="width: 10%; text-align: center;"> <ul style="list-style-type: none"> ⏪ ⏩ ⏴ ⏵ ⏴ ⏵ </div> <div style="width: 15%;"> <p>Left</p> <ul style="list-style-type: none"> Center Right </div> <div style="width: 10%; text-align: center;"> <ul style="list-style-type: none"> ⏴ ⏵ ⏴ ⏵ ⏴ ⏵ </div> <div style="width: 15%;"> <p>Group</p> <p>Add Edit Delete</p> </div> </div>		

- **tributes**
 - **avigation** : Navigation Buttons
 - **xit** : Ends or return to the previous application depending on how the "Display Detail" is configured. You can change the label within the "Button Settings".
 - **xports** : Sets the formats for printing generation:
 - **rint** : Creates a complete report with all Grid Detail data within a HTML printable page.
 - **DF** : Creates a complete report with all Grid Detail data within a PDF file type.
 - **thes**
 - **eparator** : Creates a separation line between the buttons for better display, specially for grouped buttons.

Application Hotkeys

Scriptcase allows creating shortcut keys to your applications. You can select a predefined template or create specific actions for an application.

VALUE	DESCRIPTION				
<input checked="" type="checkbox"/> <code>toggleHotkeys();</code>	Define if the application will use hotkeys				
SC_DefaultHotkeys	Select the hotkey template from previously created schemas				
Clear +					
<table border="1" style="width: 100%;"> <thead> <tr> <th>ACTION</th> <th>KEYBINDING</th> </tr> </thead> <tbody> <tr> <td colspan="2" style="text-align: center;">No hotkeys configured</td> </tr> </tbody> </table>		ACTION	KEYBINDING	No hotkeys configured	
ACTION	KEYBINDING				
No hotkeys configured					
Clear +					

Use hotkeys

Defines if the application uses hotkeys. When you enable this option, the default shortcut keys settings are disabled.

Hotkeys templat

Select the [hotkey template](#) previously created.

Action

Selects the triggered action when pressing the selected key.

Keybinding

Selects the keys responsible for executing the chosen action.

Add "+"

Adds a new action on the keys list.

Clear

It clears the selected hotkeys preference.

Mobile Toolbar

ATTRIBUTE	VALUE	DESCRIPTION
Configure the toolbar below for a "classic web version " and also for a "mobile version".		
Desktop Mobile		Copy from desktop
Top Mobile toolbar	<div style="border: 1px solid gray; padding: 5px;"> Export PDF Print Navigation Exit Separator ----- </div> <div style="display: flex; gap: 10px;"> <div style="border: 1px solid gray; padding: 5px;"> Left Center PDF Print Exit Right </div> <div style="display: flex; flex-direction: column; gap: 5px;"> ↻ ↶ ↷ ↺ ↻ </div> </div> <div style="border: 1px solid gray; padding: 5px; margin-top: 5px;"> Group Add Edit Delete </div>	
Bottom Mobile toolbar	<div style="border: 1px solid gray; padding: 5px;"> Export PDF Print Navigation Exit Separator ----- </div> <div style="display: flex; gap: 10px;"> <div style="border: 1px solid gray; padding: 5px;"> Left Center Right </div> <div style="display: flex; flex-direction: column; gap: 5px;"> ↻ ↶ ↷ ↺ ↻ </div> </div> <div style="border: 1px solid gray; padding: 5px; margin-top: 5px;"> Group Add Edit Delete </div>	

It has the same options of **Desktop** version, by adding only the item "copy of the desktop", in which, when clicked, performs a copy of toolbar items from **Desktop** to **Mobile**.

Button Settings

ATTRIBUTE	VALUE
Display Header	<input checked="" type="checkbox"/>
Detail Title	<input type="text" value="{lang_othr_detl_tit} - customers"/>
Header Variables	
SC_HEADER	<input type="text"/>
SC_VALUE	<input type="text"/>

Button Displays the buttons available in the application.

Label Allows you to customize the buttons name from current application. You can also use the option "Application Language" (placed within the main menu "Locales") to change the button labels for the whole project.

Hint Allows you to set a hint for the buttons. The hint will help the user to understand the button's action.

Shortcut key Allows you to set a shortcut key to run the button action. Important note: Each browser has its own shortcut combinations and reserved words, you need to check it before implementing this option.


Header

In this section is where the content variables that will be part of the header is set.

ATTRIBUTE	VALUE
Display Header	<input checked="" type="checkbox"/>
Detail Title	<input type="text" value="{lang_othr_detl_tit} - customers"/>
Header Variables	
SC_HEADER	<input type="text"/>
SC_VALUE	<input type="text"/>

This screen will change depending on the format chosen header within.

- **isplay Header** : This flag determines whether the header is displayed.
- **etail Title** : Using this option you can set a title for the application, it uses bring a value from the Grid application as default. However it can be customized.
- **header Variables** : Variable fields can be filled with any of the types displayed in the Combo box. Depending on the type, you must associate a content. Below we describe the existing types:
 - **ield** : When the " __Category" type option is chosen, it will open a combo box next to the fields that are part of the "Select". By choosing one of these fields, you are associating the value of the field to display in the header.
 - **itle** : When this option is selected, it will display the value filled in " **Detail Title**" to the header.
 - **ate** : When you select the " **Date**" type, the system date in the format mm/dd/yyyy will be displayed in the header. There are several display formats using server date and time. The format can be reported in the text field that appears next. Click in the question mark to check out the available formats.
 - **mage** : When the image type field is selected, a field for filling in the image name on the server will appear. To locate the existing images and select one, click on the " **Choose Image**"

icon and to make new images available on the server click on **Download an image** 

- o **alue** : When **Value** type is selected, the content filled in the text field next to that will be displayed in the header, texts and **__ Global variables__** can be entered. Ex: "Name of the Employee: [v_name]".

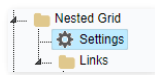
Nested grids are used when you need to display hierarchical data, such as customers and their respective applications.

The main Grid encapsulates other Grids, showing in each of its NestedGrids additional data records. Note the image below in which for each customer is displayed a Nested Grid with its applications.

LIST - CUSTOMERS				01/30/2018
Quick search <input type="text"/>				Export <input type="button" value="↓"/> Settings <input type="button" value="⚙"/>
Customerid	Companyname	Contactname	Country	
ALFKI	Alfreds Futterkiste	Maria Anders s	DE	
Orderid	Customerid	Employeeid	Orderdate	Priceorder
10,643	ALFKI	6	08/25/2008	R\$ 1.321,32
10,692	ALFKI	4	10/03/2008	R\$ 1.064,80
10,702	ALFKI	4	10/13/2008	R\$ 399,30
10,835	ALFKI	1	01/15/2008	R\$ 1.029,71
10,952	ALFKI	1	03/16/2008	R\$ 595,32
11,011	ALFKI	3	04/09/2008	R\$ 1.161,60
11,087	ALFKI	1	05/26/2009	R\$ 47,19
Grand Summary				R\$ 5.619,24
ANATR	Ana Trujillo Emparedados y helados	Ana Trujillo	MX	
ANTON	Antonio Moreno Taquerilla	Antonio Moreno	AR	

Settings

In the application menu, by clicking on the folder NestedGrid > Settings, the general attributes of the NestedGrids can be changed.



Nested Grid Settings		
ATTRIBUTE	VALUE	DESCRIPTION
Title in the same line	No	Nested Grid title line will be shown on the same line of the application title.
PDF	Yes	Defines if the Nested Grids will appear in the PDF report
XLS	Yes	It defines if the Nested Grids will appear in the XLS report
XML	Yes	It defines if the Nested Grids will appear in the XML report
Enable TreeView	No	Show the Nested Grid in TreeView mode.
Position	In one column	Position of the Nested Grid.

- **itle in the same line** : This attribute enables you to configure the Nested Grid title display. If it is displayed on the same line, the Nested Grid table will be incorporated into the main Grid.

![[Example using Nested Grid title on the same line]][nestedgrid_settings_title_same_line]

- **DF** : This option allows you to enable/disable the Nested Grid display in PDF reports.
- **LS** : This option allows you to enable / disable the Nested Grid display in the **XLS** and **XLSX** reports.
- **ML** : This option allows you to enable/disable the Nested Grid display in XML reports.
- **nable TreeView** : This option enables an option to hide/display the Nested Grid within the main Grid records. By choosing "Yes", the option "Title in the same line" will be automatically disabled.

Customerid	Companyname	Contactname	
▲ BOLID	Bolido Comidas preparadas	Martin Sommer	
Orderid	Employeeid	Orderdate	Priceorder
10,326	4	10/10/2008	R\$ 1.195,48
10,801	4	12/29/2008	R\$ 4.888,40
10,970	9	03/24/2008	R\$ 338,80
Grand Summary			R\$ 6.422,68

- **osition** : This option allows you to configure the Nested Grid placement in the main Grid. This option is not available if you enable "Title in the same line". You will be able to position the Nested Grid within the records. The options are:

○ **n one column**

Customerid	Companyname	Contactname	Orders																				
BLONP	Blondesdssl pere et fils	Frederique Citeaux	▶																				
BOLID	Bolido Comidas preparadas	Martin Sommer	▲ <table border="1"> <thead> <tr> <th>Orderid</th> <th>Employeeid</th> <th>Orderdate</th> <th>Priceorder</th> </tr> </thead> <tbody> <tr> <td>10,326</td> <td>4</td> <td>10/10/2008</td> <td>R\$ 1.195,48</td> </tr> <tr> <td>10,801</td> <td>4</td> <td>12/29/2008</td> <td>R\$ 4.888,40</td> </tr> <tr> <td>10,970</td> <td>9</td> <td>03/24/2008</td> <td>R\$ 338,80</td> </tr> <tr> <td colspan="3">Grand Summary</td> <td>R\$ 6.422,68</td> </tr> </tbody> </table>	Orderid	Employeeid	Orderdate	Priceorder	10,326	4	10/10/2008	R\$ 1.195,48	10,801	4	12/29/2008	R\$ 4.888,40	10,970	9	03/24/2008	R\$ 338,80	Grand Summary			R\$ 6.422,68
Orderid	Employeeid	Orderdate	Priceorder																				
10,326	4	10/10/2008	R\$ 1.195,48																				
10,801	4	12/29/2008	R\$ 4.888,40																				
10,970	9	03/24/2008	R\$ 338,80																				
Grand Summary			R\$ 6.422,68																				
BONAP	Bon appart	Laurence Lebihan	▶																				
BOTTM	Bottom-Dollar Markets	Elizabeth Lincoln	▶																				

○ **elow the record**

Customerid	Companyname	Contactname	
▲ BOLID	Bolido Comidas preparadas	Martin Sommer	
Orderid	Employeeid	Orderdate	Priceorder
10,326	4	10/10/2008	R\$ 1.195,48
10,801	4	12/29/2008	R\$ 4.888,40
10,970	9	03/24/2008	R\$ 338,80
Grand Summary			R\$ 6.422,68

- **ignment** : Allows you to set the Nested Grid alignment in the main Grid interface when "Position" is set set as Below the record. The options are: left, right and Center.

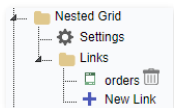
Set Individually

This is one of the values available for the attributes of the **Nested Grid Settings**

When selecting this option, the settings of attributes must be performed individually on each of the existing **Nested Grid**.

Links

To create a Nested Grid link, just click on the "New link" within the Link folder.



Creating a new Nested Grid

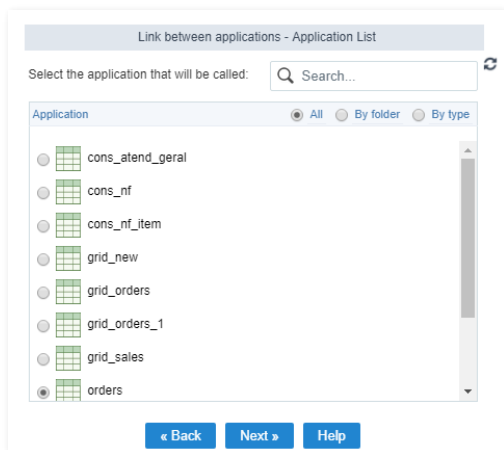
- **dding a new Nested Grid** : Enter the name and the label for the link that will be created.

Include Nested Grid	
NAME	LABEL
orders	Orders
<input type="button" value="Create"/> <input type="button" value="Help"/>	

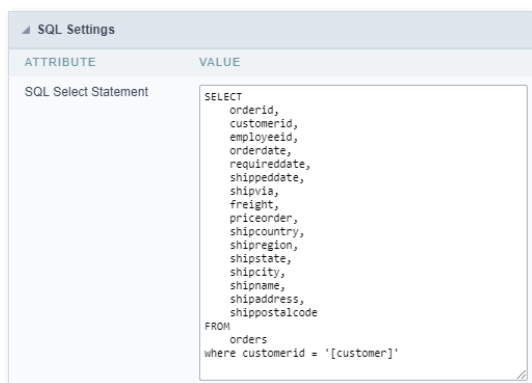
Adding a new Nested Grid

- **Application list** : Have three options for list the applications: All, by folder or by type.
 - **All** - List all applications of the current project;
 - **y Folder** - List all applications of current project, but the applications are separated respecting the project folder structure.
 - **y type** - List all applications of current project, but separated by type.

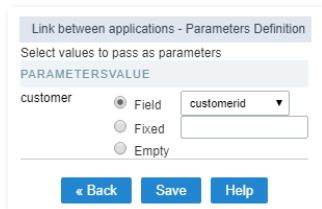
You can also search by application name in the search field.



- or an application to be used as a Nested Grid, it must have within its SQL command (Grid > SQL) a WHERE clause with a global variable to receive the parameters from the main Grid.



- **Parameters setting** : Setting the value that will be passed to the variable(s) of the Nested Grid SQL statement.

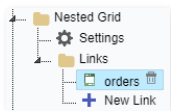


In the screen above, on the left side, it is displayed the Nested Grid input parameters (global variables created within the Nested Grid WHERE clause), on the right side, you must select the parameter option. The options are:

- **tributes**
 - **ield** : Used to assign a Grid field value as a parameter.
 - **ix** : Used to assign a fixed value as a parameter.
 - **mpty** : By selecting this option, no value will be assigned as parameter.

Editing an existing Nested Grid link

All Nested Grid links created will be placed within its menu options, within the "Links" folder. By clicking on the Nested Grid name there will be an interface for settings editing.



Set Individually

This is one of the values available for the attributes of the **Nested Grid Settings** If this value was set for some attribute in the Nested Grid settings, the settings will be made in the Nested Grid itself.

ATTRIBUTE	VALUE	DESCRIPTION
Label	Orders	Field title.
Link	<div style="border: 1px solid black; padding: 5px;"> <p>Link data</p> <p>Linked to: orders</p> <p>Parameters customer = {customerid}</p> <p>Edit link Save</p> </div>	Edit Embed Grid Link
Display	<input checked="" type="checkbox"/> Header <input checked="" type="checkbox"/> Sequential <input checked="" type="checkbox"/> Total <input type="checkbox"/> Title in the same line	Defines which items of the embed application will be shown in the main application.
Position	In one column	Position of the Nested Grid.
Enable TreeView	<input type="checkbox"/>	Show the Nested Grid in TreeView mode.
PDF	<input type="checkbox"/>	Defines if the Nested Grids will appear in the PDF report
Visualization	<input type="checkbox"/>	Defines which embed application items will inherit the visual formatting of the main application

General Settings

ATTRIBUTE	VALUE	DESCRIPTION
Label	Orders	Field title.
Link	<div style="border: 1px solid black; padding: 5px;"> <p>Link data</p> <p>Linked to: orders</p> <p>Parameters customer = {customerid}</p> <p>Edit link Save</p> </div>	Edit Embed Grid Link
Display	<input checked="" type="checkbox"/> Header <input checked="" type="checkbox"/> Sequential <input checked="" type="checkbox"/> Total	Defines which items of the embed application will be shown in the main application.
Visualization	<input checked="" type="checkbox"/>	Defines which embed application items will inherit the visual formatting of the main application

Nested Grid general settings

This interface can be used to edit the following Nested Grid attributes:


- **abel** : Title for the Nested Grid application.
- **ink** : In this attribute are displayed the current Nested Grid connection data, showing the application and parameters used. To change any link attribute just click to edit Link.
- **isplay** : Defines which items of the embedded application will be shown in the main application, that if the nested application uses some or any of these resources.
 - **eader** : Allows to display field labels.
 - **equential** : Allows to display sequential display(number line).
 - **otal** : Allows to display the Totals.
- **isualization** : When marked it sets that the Nested Grid view will be equal to that of the main Grid.

Display Settings

This settings will only display if all the following conditions are true.

Conditions of display for the **Display Settings**

Attribute	Value
Title in the same line	No
Enable Tree view	No
Position	In one column

Display Settings		
Horizontal Alignment	<input type="text"/>	Field horizontal alignment.
Vertical Alignment	<input type="text"/>	Field vertical alignment.
Background Color	<input type="text"/> 	Background Color
Title Horizontal Alignment	<input type="text"/>	Title horizontal alignment.
Title Vertical Alignment	<input type="text"/>	Title vertical alignment.
Bold	<input type="checkbox"/>	Display the field title in bold.

- **orizontal Alignment** : This option sets the Nested Grid horizontal alignment. It can be aligned left, right or Center.
- **ertical Alignment** : This option sets the Nested Grid vertical alignment. It can be aligned to the top, middle, or End.

- **background color** : Here you can set the background color. If it is not filled it will get the Grid default background color.
- **itle horizontal alignment** : This option sets the Nested Grid label horizontal alignment. It can be aligned Left, Right or Center.
- **itle vertical alignment** : This option sets the Nested Grid label vertical alignment. It can be aligned to the top, middle, or End.
- **old** : This option sets the Nested Grid title (label) in bold.

▲ SORTING RULES CONFIGURATION

ATTRIBUTE	VALUE
Sorting rules sort	<div style="border: 1px solid #ccc; padding: 5px; display: inline-block;"> Order ▲ <div style="position: absolute; right: -10px; top: 50%; transform: translateY(-50%);"> ▲ ▼ </div> </div>

Integrate sorting options Sorting Fields/Rules ▼

- **Attributes**
 - **Sorting rules sorting** : Allows you to change the sort display of the sorting rules. To configure this option, an icon is displayed (set up ordering) that when clicked, allows to select the field and your display order ascending (ASC) or descending (DESC) .
 - **Integrate sorting rules** : This option allows you to set whether to use the sorting rules together with the regular sorting options or not. You can choose the display options between:
 - **Sorting Fields/Rules(default)** : This option allows you to apply the rules that have been created together with the regular fields sorting (descending or ascending)
 - **Sorting Fields** : This option allows you to apply the fields sorting only, discarding the sorting rules (descending or ascending)
 - **Sorting Rules** : This option allows you to apply the sorting only according to the rules that have been created, discarding the fields sorting.

Creating a Sorting Rule

New Sorting Rule

Name

Label

Create

- **Attribute**
 - **name** : Field to inform the new rule name.
 - **label** : Field to inform the name that will be displayed in the application.

Setting up a sorting Rule

New Sorting Rule

Name

Label

Create

- **Attributes**
 - **label** : This option allows you to enter a name that will be displayed when the application is executed.
 - **Select the fields for a sorting rule** : Allows you to select through the selection bar which fields will be displayed (fields that are in the frame at the right side) and the fields will not be displayed (fields that are in the left side frame). To sort the fields the way you want, use the sort bar located on the right side of the frame. To apply the sorting type to the field, simply select the straight side frame field and select the Ascending mode, where displays a "+" or a "-" by the field side to indicate if the sort order is Descending.

The refined search is a feature where you can integrate a search interface, next to the Grid, limiting values according to a universe that exists in the connected database.

LIST - CUSTOMERS 04/05/2017

Quick search Columns Sorting Export Advanced Search JavaScript Link PHP Run Exit

Customerid	Companyname	Contactname	Contacttitle	Birthdate	Country
ALFKI	Alfreds Futterkiste	Maria Anders	Sales Representative	07/27/1974	DE
ANATR	Ana Trujillo Emparedados y helados	Ana Trujillo	Owner	02/10/1976	MX
ANTON	Antonio Moreno Taquería	Antonio Moreno	Owner	03/04/1977	AR
AROUT	Around the Horn	Thomas Hardy	Sales Representative	10/10/1974	GB
BERGS	Berglunds snabbkop	Christina Berglund	Order Administrator	01/22/1976	SE
BLAUS	Blauer See Delikatessen	Hanna Moos	Sales Representative	05/11/1975	DE
BLONP	Blondesdélis père et fils	Frédérique Citeaux	Marketing Manager	03/07/1978	FR
BOLID	Bolido Comidas preparadas	Martin Sommer	Owner	11/11/1978	ES
BONAP	Bon appart	Laurence Leblan	Owner	05/16/1975	FR
BOTTM	Bottom-Dollar Markets	Elizabeth Lincoln	Accounting Manager	01/08/1975	CA

Go to 1 View 10 [1 to 10 of 91]

Grid application running with refined search activated

Settings

REFINED SEARCH SETTINGS

ATTRIBUTE	VALUE
Move searched above	<input type="checkbox"/>
Show quantity	<input checked="" type="checkbox"/>
Start mode	Start open
Minimum width	260px
Maximum width	
Minimum height	
Maximum height	

Refined search settings

- Attributes**
 - move searched above** : When you filter by a field, this field will be moved to the top of the refined search stack.
 - show quantity** : Alongside the values of the fields, you will see the existing amount. Ex.: Brazil (1547)
 - start mode** : This option sets whether the fields select options will start open or closed.
 - minimum width** : This option sets the minimum width for the refined search fields, value in pixels.
 - maximum width** : This option sets the maximum width for the refined search fields, value in pixels.
 - minimum height** : This option sets the minimum height for the refined search fields results, value in pixels.
 - maximum height** : This option sets the maximum height for the refined search fields results, value in pixels.

Select fields

Refined Search Fields

The refined search is automatically added to the Grid application when one or more Grid fields are selected in the tab "select fields".

REFINED SEARCH FIELDS

Select Fields Edit Fields

Select the fields that will part of the refined search. Use the tab "Edit Fields" to configure the settings.

- birthdate
- creditlimit
- country
- customerid
- companyname
- contactname
- contacttitle
- regionid
- stateid
- city
- address
- postalcode
- phone
- fax
- cityid
- cardtype
- cardnumber
- notes

Edit fields

In the tab "Edit fields", you can configure each field according to what you need to display within the application.

REFINED SEARCH FIELDS

Select Fields Edit Fields

Fields	Range			Multiselect	Start opened	See more	Quantity	Date	Sorting	Message for empty fields
	Range	Increment	Show value							
birthdate				<input checked="" type="checkbox"/>	<input type="text" value="v"/>	<input type="checkbox"/>		ddmmyyyy	ASC	{lang_refine_search_empty}
creditlimit	<input checked="" type="checkbox"/>	<input type="text" value="5"/>	<input checked="" type="checkbox"/>		<input type="text" value="v"/>				ASC	{lang_refine_search_empty}
country				<input checked="" type="checkbox"/>	<input type="text" value="v"/>	<input checked="" type="checkbox"/>	<input type="text" value="10"/>		ASC	{lang_refine_search_empty}

Refined Search fields editing

- **tributes**
 - **ields** : Listing of selected fields in the tab "select fields".
 - **ange** : This option activate a slider with intervals between the smallest and largest value.
 - **crement** : When using the range this option will set an integer value to increase the range.
 - **how value** : When checked, displays the range values in the slider.
 - **ulti select** : Enables multiple selection values to perform the filter.
 - **tart opened** : This option sets the initial mode for fields selection display in refined search.
 - **ee more** : When checked, enables the option to "see more" for the filter records. This is recommended when the selected field has a big amount of records. You will be able to click to "see more" or "see less"
 - **uantity** : This option sets the initial amount of records to be displayed for each field selected.
 - **ate** : This option sets the date format that is displayed in the date fields inside the filter.
 - **orting** : This option sets the records sorting for each selected field
 - **essage for empty fields** : This option sets a message to be displayed when the field has empty values. You can also use a variable from Scriptcase language system.

Group By Settings

These are the general settings of the "Group By" in the Grid, independently of its type, static or dynamic. Here we can define some Group By viewing options, such as the use of Tree view, for example.

GROUP BY SETTINGS	
ATTRIBUTE	VALUE
Enable TreeView	<input type="checkbox"/>
Group By Line	Before the records ▾
Group By Header	<input type="checkbox"/>
Group By sorting	<input checked="" type="checkbox"/>
Display Titles	<input checked="" type="checkbox"/>
Record count title	<input type="text"/>
Value Separator	<input type="text" value="=>"/>
Tab a Group By	<input type="text" value="10px"/>
Separates the Group By	<input type="text" value="10px"/>

*The option **Enable TreeView** isn't available in the Infinite Scroll.*

- **Enable TreeView** : Defines the use of TreeView in "Group By", which allows you to expand or collapse the records displayed in Group by.
- **Group By Line** : Sets the positioning of group by in relation to group records. The options are **Before the records** or **After the records**.
- **Group By Header** : Lets you display the Group By header on all pages.
- **Group By sorting** : It allows to order the fields respecting the Group By.
- **Display Titles** : Allows the label display of the fields within the groups.
- **Record count title** : Sets a title for the records quantity column.
- **Value Separator** : Define the separator between the title and the field value in the GROUP BY.
- **Tab a Group By** : Sets the left margin of Group By. The value must be informed in pixel.
- **Separates the Group By** : Sets the spacing between two Group By. The value must be informed in pixel.

Dynamic Group By

In this screen we define the fields available in Dynamic Group By and will be summarized in Grid and Summary when the Dynamic Group By is used.

Select Fields

In this screen we define the fields available in the Dynamic Group By and will be summed in the Grid and Summary when the dynamic Group By is used.

This Group By works independently from any other Group By configuration or Totals already performed, that is, the fields defined to be summarized in the Grid or Summary in the Dynamic Group will be visible only when the Dynamic Group is used.

We will be able to see all available fields for the Group By configuration and totals in the **Grid Fields**.

The **Use Dynamic Group By** option enables the Group By in the run-time application.

Fields used when using the Grid must be configured to be displayed on the Grid.

GROUP BY

Use dynamic Group By

GRID FIELDS	GROUP BY FIELDS	GRID TOTALS	SUMMARY TOTALS
<ul style="list-style-type: none"> T customerid T companyname T contactname T contacttitle 📅 birthdate T country # regionid T stateid T city T address T postalcode T phone T fax # cityid \$ creditlimit T cardtype T cardnumber T notes # Record Count 			

- **Grid Fields**: List of all the fields of the application.
- **Group By Fields**: Defines the fields that are part of the Group By.
- **Grid Totals**: Defines the fields that are part of the Grid Totals.
- **Summary Totals**: Defines the fields that are part of the Summary Totals.

Group By Fields

We must drag and drop the fields that will be part of the Group By in **Group By Fields**.

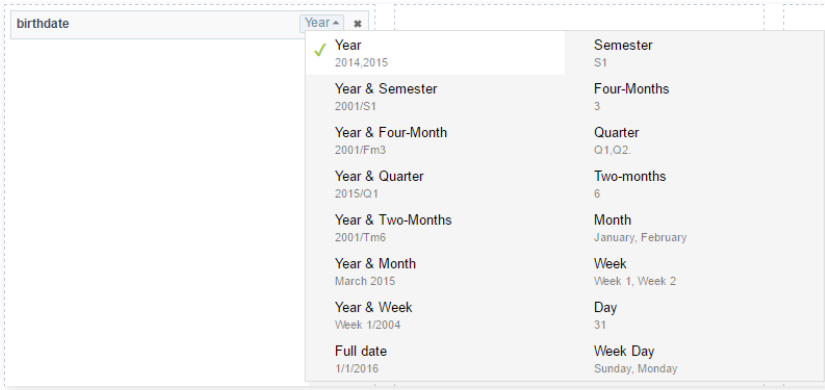
GROUP BY FIELDS

country		✕
regionid		✕
birthdate	Year	✕
birthdate	Semester	✕

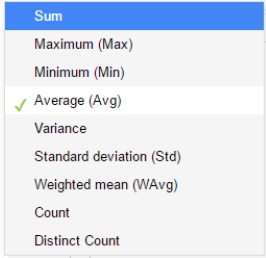
Each field can only be added once to the **Group By Fields**, except for the date and datetime fields.

Date and Datetime fields

For date and datetime fields, some display intervals have been added, so these fields can be added two or more times to the Group By.



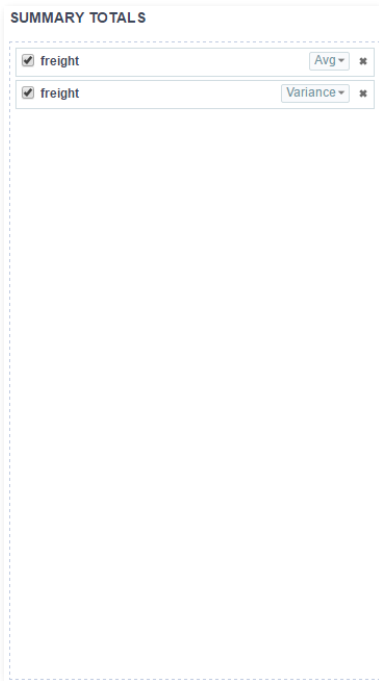
All Grid fields can be added to the “totals”, the only difference being the functions available. Numeric fields can use all the available summarization functions, since the non-numeric fields can use only the count and the different count.



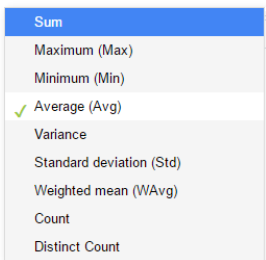
The fields configured in this totals can be edited in **Grid > Totals > Fields > Dynamic Group By**

Summary Totals

The fields added to the Summary totals will only be visible in Summary at the time Dynamic Group By is used.



All Grid fields can be added to the “totals”, the only difference being the functions available. Numeric fields can use all the available summarization functions, since the non-numeric fields can use only the count and the different count.



The fields configured in this totals can be edited in **Summary > Fields > Dynamic Group By**

Fields

The fields added to the **Grid Fields** will be listed within the Fields folder so that they can be edited individually.

General Settings

Categories: Text and Barcode

GENERAL SETTINGS: CUSTOMERID	
ATTRIBUTE	VALUE
Data Type	Text
Group by label	
Case Settings	
Field Mask	
SQL Type	VARCHAR

- **Data Type:** Defines the data type of the field in the HTML.
- **Group by label:** Defines a label of the fields displayed in the Group By.
- **Case Settings:** Converts the text according to the selected option.
 - **Upper Case:** Transforms the text in upper case.
 - **Lower Case:** Transforms the text in lower case.
 - **Capitalize first word:** It transforms the first word in the upper case or lower case, according to the text retrieved, that is, if the text retrieved by the application is in lower case, this option will make upper case only the first word.
 - **Capitalize all words:** Transforms all words in the upper case or lower case, according to the text retrieved, that is, if the text retrieved by the application is in lower case, this option will transform upper case into all words.
- **Field Mask:** On this field you will configure the display mask according to the table informed in Applications > Grid > Fields > Text.
- **SQL Type:** Informs the data type of the field in the database.

Categories: Number and Calculated

GENERAL SETTINGS: EMPLOYEEID	
ATTRIBUTE	VALUE
Data Type	Integer
Group by label	
Field Mask	
SQL Type	INT

- **Data Type:** Defines the data type of the field in the HTML.
- **Group by label:** Defines a label of the fields displayed in the Group By.
- **Field Mask:** On this field you will configure the display mask according to the table informed in Applications > Grid > Fields > Text.
- **SQL Type:** Informs the data type of the field in the database.

Categories: Special and Date/Time

GENERAL SETTINGS: ORDERDATE	
ATTRIBUTE	VALUE
Data Type	Date
Group by label	{lang_othr_cons_title_YYYY}
	Year of %s
Label of the values	{lang_othr_valueYYYY}
SQL Type	DATE

- **Data Type:** Defines the data type of the field in the HTML.
- **Group by label:** Defines a label of the fields displayed in the Group By.
- **SQL Type:** Informs the data type of the field in the database.

Lookup Settings

This option is available in the fields of the categories: Text, Number, Calculated, Special and Barcode.

On this option, it's possible to setup the display lookup for the fields cited above. For more information about using the Grid Lookup, access Application > Grid > Fields > Text.

Values Format


Number Type

VALUES FORMAT	
ATTRIBUTE	VALUE
Regional Settings	<input checked="" type="checkbox"/>
Color of Negative	<input type="text"/>

- **Regional Settings:** When active, you can apply the Regional Settings to this field. To modify the setting, access Locales > Regional Settings in the Scriptcase Menu.

- **Color of Negative:** In this field you'll inform the color value in hexadecimal. (Example: #000000)

Currency Type

VALUES FORMAT	
ATTRIBUTE	VALUE
Regional Settings	<input checked="" type="checkbox"/>
Currency Format	<input type="checkbox"/>
Color of Negative	<input type="text"/> 
Decimal Precision	<input type="text" value="0"/>
Complete with Zeros	<input checked="" type="checkbox"/>

- **Regional Settings:** When active, you can apply the Regional Settings to this field. To modify the setting, access Locales > Regional Settings in the Scriptcase Menu.
- **Currency Symbol:** When this option is activated, the application will display the currency symbol according to the Regional Settings.
- **Color of Negative:** In this field you'll inform the color value in hexadecimal. (Example: #000000)
- **Decimal Precision:** Amount of decimal that your field will be displaying.
- **Complete with Zeros:** Activating this option will allow the application to complete the value after the coma with zeros.

Date Type

VALUES FORMAT	
ATTRIBUTE	VALUE
Regional Settings	<input checked="" type="checkbox"/>
Internal Format	<input type="text"/> ?

- **Regional Settings:** When active, you can apply the Regional Settings to this field. To modify the setting, access Locales > Regional Settings in the Scriptcase Menu.
- **Display:** In this field you will select how will the field will display itself.

Group By Settings

- **Attributes**
 - **Field Position :** Defines how the information contained in the Group By line will be displayed and organized.
 - **Columns :** Defines the amount of columns information contained in the Group By line will displayed and organized.
 - **Display Label :** Defines if the field label will be displayed.
 - **Line break :** Displays or not the Group By line with the value divided in another line.
 - **Records amount :** Defines whether the Group By row will display the number of group records.
 - **Break PDF Page (Grid) :** Defines if the PDF file, generated by the GRID, will contain each group by printed in a new page.
 - Ex. In a report you can have some orders where they can be displayed on different pages.
 - **Break PDF Page (Summary) :** Defines if the Summary PDF file, generated by the GRID will contain each Group By printed in a new page.
 - Ex. In a report you can have some orders where they can be displayed on different pages. .
 - **Break HTML Page (Grid) :** Displays each Grid Group By in a different HTML page.
 - **Break HTML Page (Summary) :** Displays each Summary Group By in a different HTML page.
 - **Start TreeView :** Defines the initial state of the TreeView.
 - **Sorting :** If this option is set to "Yes", the selected fields will sorted without the need to be clicked on, the sorting will be done respecting the criteria of the Group By.
 - **Fields :** Defines which fields are going to displayed in the Group By row. It's also possible to define if it's going to be displayed in the Totals or in the Summary of this field by selecting Value or Sum, before clicking on the button "On/Off".

Group By Line Layout (Label)

Layout configuration of the Group By Label.

- **Attributes**
 - **Font Style :** Defines the font that's going to be used in the Label.
 - **Font Size :** Defines the size of the font used in the Group By Label.
 - **Font Color :** Font color for the Group By Label.
 - **Background Color :** Group By background Color.
 - **Bold Text :** Defines if the label will have a Bold Text.

Group By Line Layout (Value)

Layout configuration of the Group By Value.

- **Attributes**
 - **Font Style :** Defines the font that is going to be used in the Value.
 - **Font Size :** Defines the size of the font used in the Group By Value.
 - **Font Color :** Font color for the Group By Value.
 - **Bold Text :** Defines if the value will have a Bold Text.

Static Group By

This type of Group By is configured by the developer, where the end user can select one of the predefined Group By available in the application.

Settings

These configurations affects the Static Group By.

Static Group By Settings

ATTRIBUTE	VALUE
Use empty Group By	<input checked="" type="checkbox"/>
Title of the empty Group By	<input type="text" value="(lang_othr_groupby_none)"/>
Initial Group By	<input type="text"/>
Use dynamic totalization	<input checked="" type="checkbox"/>

- **Use empty Group By** : Configuration used when you want the Grid to be initialized without any Group By.
- **Title of the empty Group By** : Defines a title for the empty Group By option. (Available only when the previous option is enabled)
- **Initial Group By** : Defines the Group By used when running the generated application.

Sorting Groups Settings

Defines the order that the Group By's will be displayed in the Grid.

ATTRIBUTE	VALUE
Group By sorting	<input type="text" value="Rule1"/>

New Group By

Interface to create the static Group By.

GROUP BY			
Name <input type="text"/>		Label <input type="text"/>	
GRID FIELDS	GROUP BY FIELDS	GRID TOTALS	SUMMARY TOTALS
<ul style="list-style-type: none"> # orderid T customerid # employeed # orderdate # requireddate # shippeddate # shipvia \$ freight \$ priceorder T shipcountry # shipregion T shipstate # shipcity T shipname T shipaddress T shippostalcode # Record Count 			

- **Name**: Defines the internal name of the Group By, used by ScriptCase.
- **Label**: Defines the displayed name of the Group By, that will be displayed in the application.
- **Grid Fields**: List of all the fields of the application.
- **Group By Fields**: Defines the fields that are part of the Group By.
- **Grid Totals**: Defines the fields that are part of the Grid Totals.
- **Summary Totals**: Defines the fields that are part of the Summary Totals.

Group By Fields

You need to drag and drop the fields that'll be part of the Group By in the **Group By Fields**.

GROUP BY FIELDS

customerid	Eixo Y	*
orderid	Eixo Y	*
employeeid	Eixo Y	*
orderid	Year	Eixo Y *
orderid	Semester	Eixo Y *

Each field can only be added once to the **Group By Fields**, except for the date and datetime fields.

Date and Datetime fields

For date and datetime fields, some display intervals have been added, so these fields can be added two or more times to the Group By.

birthdate Year *

<ul style="list-style-type: none"> ✓ Year 2014, 2015 Year & Semester 2001/S1 Year & Four-Month 2001/Fm3 Year & Quarter 2015/Q1 Year & Two-Months 2001/Tm6 Year & Month March 2015 Year & Week Week 1/2004 Full date 1/1/2016 	<ul style="list-style-type: none"> Semester S1 Four-Months 3 Quarter Q1, Q2 Two-months 6 Month January, February Week Week 1, Week 2 Day 31 Week Day Sunday, Monday
--	---

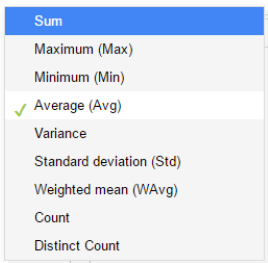
Grid Totals

The fields added in this total will be visible only in the Grid (if the total field is visible in the Grid) at the time that the group by is used.

GRID TOTALS

priceorder	Avg	*
priceorder	Variance	*
orderid	Count	*

All Grid fields can be added to the "totals", the only difference being the functions available. Numeric fields can use all the available summarization functions, since the non-numeric fields can use only the count and the different count.



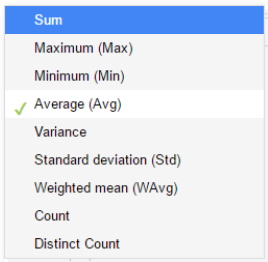
The fields configured in this totals can be edited in **Grid > Totals > Fields > Dynamic Group By**

Summary Totals

The fields added to the Summary totals will only be visible in Summary at the time Dynamic Group By is used.



All Grid fields can be added to the "totals", the only difference being the functions available. Numeric fields can use all the available summarization functions, since the non-numeric fields can use only the count and the different count.



The fields configured in this totals can be edited in **Summary > Fields > Dynamic Group By**

Edit Group By

After the creation steps, all the Group By will be listed in the **Static Group By** folder, below the Settings item.

Expanding the Group By Folder, you will see the **settings** icon, where you can edit the entire Group By.

You can also view the fields used in the Group By, that can be edited independently from the Grid Layout, see the following:

Fields

The fields added to the **Grid Fields** will be listed within the Fields folder so that they can be edited individually.

General Settings

Categories: [Text and Barcode](#)

GENERAL SETTINGS: CUSTOMERID	
ATTRIBUTE	VALUE
Data Type	Text
Group by label	
Case Settings	
Field Mask	
SQL Type	VARCHAR

- **Data Type:** Defines the data type of the field in the HTML.
- **Group by label:** Defines a label of the fields displayed in the Group By.
- **Case Settings:** Converts the text according to the selected option.
 - **Upper Case:** Transforms the text in upper case.
 - **Lower Case:** Transforms the text in lower case.
 - **Capitalize first word:** It transforms the first word in the upper case or lower case, according to the text retrieved, that is, if the text retrieved by the application is in lower case, this option will make upper case only the first word.
 - **Capitalize all words:** Transforms all words in the upper case or lower case, according to the text retrieved, that is, if the text retrieved by the application is in lower case, this option will transform upper case into all words.
- **Field Mask:** On this field you will configure the display mask according to the table informed in Applications > Grid > Fields > Text.
- **SQL Type:** Informs the data type of the field in the database.

Categories: Number and Calculated

GENERAL SETTINGS: EMPLOYEEID	
ATTRIBUTE	VALUE
Data Type	Integer
Group by label	
Field Mask	
SQL Type	INT

- **Data Type:** Defines the data type of the field in the HTML.
- **Group by label:** Defines a label of the fields displayed in the Group By.
- **Field Mask:** On this field you will configure the display mask according to the table informed in Applications > Grid > Fields > Text.
- **SQL Type:** Informs the data type of the field in the database.

Categories: Special and Date/Time

GENERAL SETTINGS: ORDERDATE	
ATTRIBUTE	VALUE
Data Type	Date
Group by label	{lang_othr_cons_title_YYYY}
	Year of %s
Label of the values	{lang_othr_valueYYYY}
SQL Type	DATE

- **Data Type:** Defines the data type of the field in the HTML.
- **Group by label:** Defines a label of the fields displayed in the Group By.
- **SQL Type:** Informs the data type of the field in the database.

Lookup Settings

This option is available in the fields of the categories: Text, Number, Calculated, Special and Barcode.

On this option, it's possible to setup the display lookup for the fields cited above. For more information about using the Grid Lookup, access Application > Grid > Fields > Text.

Values Format

Number Type

VALUES FORMAT	
ATTRIBUTE	VALUE
Regional Settings	<input checked="" type="checkbox"/>
Color of Negative	<input type="text"/>

- **Regional Settings:** When active, you can apply the Regional Settings to this field. To modify the setting, access Locales > Regional Settings in the Scriptcase Menu.
- **Color of Negative:** In this field you'll inform the color value in hexadecimal. (Example: #000000)

Currency Type

VALUES FORMAT	
ATTRIBUTE	VALUE
Regional Settings	<input checked="" type="checkbox"/>
Currency Format	<input type="checkbox"/>
Color of Negative	<input type="text"/>
Decimal Precision	<input type="text" value="0"/>
Complete with Zeros	<input checked="" type="checkbox"/>

- **Regional Settings:** When active, you can apply the Regional Settings to this field. To modify the setting, access Locales > Regional Settings in the Scriptcase Menu.
- **Currency Symbol:** When this option is activated, the application will display the currency symbol according to the Regional Settings.
- **Color of Negative:** In this field you'll inform the color value in hexadecimal. (Example: #000000)
- **Decimal Precision:** Amount of decimal that your field will be displaying.
- **Complete with Zeros:** Activating this option will allow the application to complete the value after the coma with zeros.

Date Type

VALUES FORMAT	
ATTRIBUTE	VALUE
Regional Settings	<input checked="" type="checkbox"/>
Internal Format	<input type="text"/>

- **Regional Settings:** When active, you can apply the Regional Settings to this field. To modify the setting, access Locales > Regional Settings in the Scriptcase Menu.
- **Display:** In this field you will select how will the field will display itself.

Group By Settings

- **Attributes**
 - **Field Position :** Defines how the information contained in the Group By line will be displayed and organized.
 - **Columns :** Defines the amount of columns information contained in the Group By line will displayed and organized.
 - **Display Label :** Defines if the field label will be displayed.
 - **Line break :** Displays or not the Group By line with the value divided in another line.
 - **Records amount :** Defines whether the Group By row will display the number of group records.
 - **Break PDF Page (Grid) :** Defines if the PDF file, generated by the GRID, will contain each group by printed in a new page.
 - Ex. In a report you can have some orders where they can be displayed on different pages.
 - **Break PDF Page (Summary) :** Defines if the Summary PDF file, generated by the GRID will contain each Group By printed in a new page.
 - Ex. In a report you can have some orders where they can be displayed on different pages. .
 - **Break HTML Page (Grid) :** Displays each Grid Group By in a different HTML page.
 - **Break HTML Page (Summary) :** Displays each Summary Group By in a different HTML page.
 - **Start TreeView :** Defines the initial state of the TreeView.
 - **Sorting :** If this option is set to "Yes", the selected fields will sorted without the need to be clicked on, the sorting will be done respecting the criteria of the Group By.
 - **Fields :** Defines which fields are going to displayed in the Group By row. It's also possible to define if it's going to be displayed in the Totals or in the Summary of this field by selecting Value or Sum, before clicking on the button "On/Off".

Group By Line Layout (Label)

Layout configuration of the Group By Label.

- **Attributes**
 - **Font Style :** Defines the font that's going to be used in the Label.
 - **Font Size :** Defines the size of the font used in the Group By Label.
 - **Font Color :** Font color for the Group By Label.
 - **Background Color :** Group By background Color.
 - **Bold Text :** Defines if the label will have a Bold Text.

Group By Line Layout (Value)

Layout configuration of the Group By Value.

- **Attributes**
 - **Font Style :** Defines the font that is going to be used in the Value.
 - **Font Size :** Defines the size of the font used in the Group By Value.
 - **Font Color :** Font color for the Group By Value.
 - **Bold Text :** Defines if the value will have a Bold Text.

Events

These events are only available for static Group By.

The **OnGroupByAll** event occurs after running the Group By, that allows you to manipulate the totals variables.

Total variables are created based on the fields selected for group by and totals.

The following is an example of the available variables:

Assuming that an application has a Group By **state** and **city** and two summarization per **parcel** and **balance**, the following summarization variables are available:

{count_ger} Contains the total number of records.
{sum_parcel} Contains the general sum of the parcel field.
{sum_balance} Contains the overall sum of the balance field.
{count_state} Contains the total amount of records, from the state group by being processed.
{sum_parcel_state} Contains the general sum of the 'state' field and the 'parcel' field that are being processed.
{sum_balance_state} Contains the general sum of the 'balance' field and the 'parcel' field that are being processed.
{count_city} Contains the total number of records, of the city group by being processed.
{sum_parcel_city} Contains the general sum of the 'parcel' and 'city' field that are being processed.
{sum_balance_city} Contains the general sum of the 'balance' and 'city' fields that are being processed.

To access the summarization variables by group, simply replace the Group By name with **Group By**. Example: **{sum_balance_city}** for **{sum_quebra_balance}**

{count_quebra} The total number of records in the Group By that is being processed.
{sum_quebra_parcel} Contains the general sum of the parcel field, of the Group By that is being processed.
{sum_quebra_balance} Contains the sum total of the balance field, of Group By that is being processed.

Example:

In an application that has Group By by state and city and totals a balance field in Group By totals, we want to display the average in place of the balance. A method is created in the **OnGroupByAll** event, with the following content:

```
{sum_quebra_balance} = {sum_quebra_balance} / {count_quebra};
```

General View

The total fields will be displayed only when the Grid application is using at least one Group By.

Settings

The settings below can be applied for the Grid totals only.

Summarization Functions ?		
ATTRIBUTE	VALUE	DESCRIPTION
Results in a single line.	<input checked="" type="checkbox"/>	Display the Main Total title and its value on a single line.
Display Total	On every page ▾	Pages where the total will be displayed.
Group Subtotal	Below ▾	Display the Group subtotal after the records of the Group By.
Record Count	<input checked="" type="checkbox"/>	Display the record count in the total.

- **results in a single line.**: This option sets the display of the **General Total** title and its results in a single line.

- sample for this option **Enabled**:

Grand Summary(872) - Sum	R\$ 267.620,19
- Max	R\$ 4.791,60
- Min	R\$ 0,00
Grand Summary (872)	
Sum	R\$ 267.620,19
Max	R\$ 4.791,60
Min	R\$ 0,00

- sample for this option **Disabled**:

- **isplay Total** : This option indicates in which pages the General Totals will be displayed. The options are: **On every page**, **On the last page** or **Do not display**. * **Group Subtotal** : This option defines where the subtotal of group by will be displayed.
- **ecord Count** : This option allows you to view the amount of records by the General Total title.
 - sample for this option **Enabled**: **Grand Summary(872)** * Example for this option **Disabled**: **Grand Summary**

Layout settings

Display settings of the group subtotal.

Layout settings ?		
reg1		
GROUP BY FIELD	LABEL	DISPLAY
date_YYYYMMDD2	Group Subtotal	<input checked="" type="checkbox"/>
reg2		
GROUP BY FIELD	LABEL	DISPLAY
region		<input checked="" type="checkbox"/>

- **abel** - This field sets the Label for the **Group Subtotal**.
- **isplay** - This option sets if the **Group Subtotal** will be displayed.

Select fields

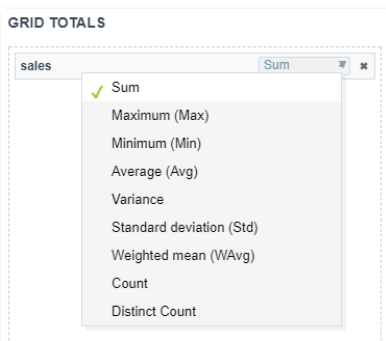
Using the select fields area you can set the total fields and the total options for each field. The same field can be used more than once in the Grid totals area, just if they are using different summaries types.

To define the fields that will be used for totals, drag them to the area, **Grid Totals**.



Note: The total fields will only be displayed if they are also selected to be displayed within the Grid application module.

When positioning the fields, you must define what summarization will be used, to select that you must click in the combo box and select one of the available options, that will be according to the data type (integer, date, text...).



The summarization options available are:

- **um** : Sets a sum of the values for the selected field.
- **aximum** : Displays the highest value identified in the selected field.
- **inimum** : Displays the lowest value identified in the selected field.
- **verage (Avg)** : Calculates the arithmetic mean of the values for the selected field.
- **ariance** : Calculates the dispersion of the values related to the average.
- **tandard Deviation** : Measures the variability of values around the average, the minimum value of the standard deviation is 0 indicating that there is variability, i.e. that all values are equal to the mean.
- **ighted mean (WAvg)** : Calculates the weighted average for the selected field. To set the weight used in the calculation of the average access field settings selected in **Totals > Fields (select the field where you are using the Weighted mean) > Weighted average weight**.
 - **ighted average weight** : Field that will be used as weighted average weight. In calculating the weighted average, each set value is multiplied by its "weight", that is its relative importance.
- **ount** : Displays the total number of records for the selected field.
- **istinct Count** : Displays the total number of records for the selected field, distinguishing the values.

Important note: The fields in the Grid totals are displayed only on Grids with no Group by or with empty Group by.

Positioning - Default

Defines the positioning and the label used by Grid totals. There are three display formats,, **Default**, **Grouped** or **By field**

To add the same field two or more times in the Totals or use different types of summaries in selected fields, this option will not be displayed.

The option **Default** returns the result below the column being summarized. When you use the total for more than one column of the Grid, using the same type of summarization, the results are displayed on the same line.

Summary Fields Order

Default
 Grouped
 By field
 Alignment: Left

Illustration

FIELD 1	FIELD 2	FIELD 3	FIELD 4	FIELD 5
001	Testing...	Testing...	\$ 7,552.96	\$ 1,759.83
002	Testing...	Testing...	\$ 97.16	\$ 22.63
003	Testing...	Testing...	\$ 5,568.00	\$ 1,297.34
Total			\$ 13,218.12	\$ 3,079.82

The line with **General Total** displaying the **Sales**.

Id	Date	Region	Product Category	Sales
1	01/01/2015	South Atlantic	SEAFOOD	R\$ 50,82
2	01/01/2016	South Atlantic	CONFECTIONS	R\$ 84,70
3	01/01/2013	South Atlantic	BEVERAGES	R\$ 90,75
4	01/02/2013	East South Central	CONFECTIONS	R\$ 163,35
5	01/02/2014	East South Central	DAIRY PRODUCTS	R\$ 251,68
Grand Summary				R\$ 267.620,19

Alignment

This option is available for display formats **Default** and **Grouped**.

Defines the placement of the label selected within the **Setting > Display total** option. The alignment can be in the **Center, Left** e **Right**:

- eft** : Grand Summary R\$ 267.620,19
- ight** : Grand Summary R\$ 267.620,19
- enter** : Grand Summary R\$ 267.620,19

Label settings (Default)

This option allows you to customize the total labels.

As default we are going to display Grant totals

Label settings (Default)

ATTRIBUTE	VALUE	DESCRIPTION
Total line message	<input type="text" value="(lang_msgs_tot)"/>	Message displayed in the line of total

Positioning - Grouped

The option **Grouped** returns the total result, separating each type per line.

It is possible to position the total lines by dragging to the desired position.

Summary Fields Order

Default
 Grouped
 By field
 Alignment: Left

FIELD 1	FIELD 2	FIELD 3	FIELD 4	FIELD 5
001	Testing...	Testing...	\$ 550.00	\$ 200.00
002	Testing...	Testing...	\$ 150.00	\$ 350.00
+ Sum			\$ 700.00	\$ 550.00
+ Average (Avg)			\$ 350.00	\$ 275.00
+ Maximum (Max)			\$ 550.00	\$ 350.00
+ Minimum (Min)			\$ 150.00	\$ 200.00
+ Count			\$ 250.00	\$ 470.00
+ Distinct Count			\$ 95.00	\$ 300.00
+ Variance			\$ 50.00	\$ 120.00
+ Standard deviation (Std)			\$ 200.00	\$ 900.00
+ Weighted mean (WAvg)			\$ 150.00	\$ 500.00

In this example, we are displaying the sum, average, maximum and minimum of column **Sales**.

Id	Date	Region	Product Category	Sales
1	01/01/2015	South Atlantic	SEAFOOD	R\$ 50,82
2	01/01/2016	South Atlantic	CONFECTIONS	R\$ 84,70
3	01/01/2013	South Atlantic	BEVERAGES	R\$ 90,75
4	01/02/2013	East South Central	CONFECTIONS	R\$ 163,35
5	01/02/2014	East South Central	DAIRY PRODUCTS	R\$ 251,68
Grand Summary				
Sum				R\$ 267.620,19
Avg				R\$ 306,90
Max				R\$ 4.791,60
Min				R\$ 0,00

Alignment

This option is available for display formats **Default** and **Grouped**.

Defines the placement of the label selected within the **Setting > Display total** option. The alignment can be in the **Center, Left e Right**:

- **eft** : **Grand Summary** R\$ 267.620,19
- **ight** : **Grand Summary** R\$ 267.620,19
- **enter** : **Grand Summary** R\$ 267.620,19

Label settings (Grouped)

This option allows you to customize the total labels.

ATTRIBUTE	VALUE	DESCRIPTION
Sum Label	<input type="text" value="{lang_btms_smry_msge_sum}"/>	Sum function title
Average Label	<input type="text" value="{lang_btms_smry_msge_avg}"/>	Average function title
Max Label	<input type="text" value="{lang_btms_smry_msge_max}"/>	Function max title
Minimum Label	<input type="text" value="{lang_btms_smry_msge_min}"/>	Min function title.
Count Label	<input type="text" value="{lang_btms_smry_msge_cnt}"/>	Title function Count
Distinct Count label	<input type="text" value="{lang_btms_smry_msge_dct}"/>	Title function Distinct Count
Variance Label	<input type="text" value="{lang_btms_smry_msge_var}"/>	Title function Variance
Standard Deviation Label	<input type="text" value="{lang_btms_smry_msge_pad}"/>	Title function Standard Deviation
Weighted mean label	<input type="text" value="{lang_btms_smry_msge_wei}"/>	Label for the weighted mean title

Positioning - By field

This option displays the values in the left corner by positioning the results next to each other.

Summary Fields Order

Grouped
 By field
 Break line per field

Illustration

FIELD A	FIELD B	FIELD C	FIELD D	FIELD E
001	Testing...	Testing...	\$ 400.00	\$ 100.00
002	Testing...	Testing...	\$ 250.00	\$ 350.00
Total				

Line break per field

This option sets the total fields display. If selected, it will show the fields in the same row or divided by line.

Separated by line:

Grand Summary

Sales (Sum) R\$ 267.620,19
 Profit (Sum) (\$803,977.68)

Displayed on the same line:

Grand Summary

Sales (Sum) = R\$ 267.620,19 Profit (Sum) = (\$803,977.68)

Label settings (By field)

This option allows you to customize the total labels.

Label settings (By field) ?			
ATTRIBUTE	ATTRIBUTE	VALUE	DESCRIPTION
Totals	Label (sales sum)	<input type="text" value="Sales ((lang_btms_smry_msge_sum))"/>	Título do campo.
	Label (profit sum)	<input type="text" value="Profit ((lang_btms_smry_msge_sum))"/>	Título do campo.

Fields

This option allows you to configure the display of values and labels of the selected fields in the Grid totals.

General Settings

Allows you to change the label displayed in the totals.

General Settings: sales_sum		
ATTRIBUTE	VALUE	DESCRIPTION
Label	<input type="text" value="Sales ((lang_btms_smry_msge_sum))"/>	Título do campo.
	Sales (Sum)	

Visual configuration for the Grid totals

Sets the formatting of the fields displayed in the subtotal for the group.

Visual settings of totalization in a Group By		
ATTRIBUTE	VALUE	DESCRIPTION
Text font	<input type="text" value="Aa"/>	Text field font
Font Size	<input type="text" value="v"/>	Font Size
Font Color	<input type="text" value="c"/>	Font Color
Background Color	<input type="text" value="c"/>	Background Color
Text bold.	<input type="text" value="v"/>	Formatting text bold.

- **ont family** : Sets the font used.
- **ont size** : Sets the font size.
- **ext color** : Sets the text color
- **ackground color** : Sets the background color.
- **old** : Format text in bold.

Example of formatting the subtotal for the Group:

Date of Date => 01/09/2013					
Id	Date	Region	Product Category	Sales	Profit
22	01/09/2013	East South Central	BEVERAGES	R\$ 163,35	\$36.30
19	01/09/2013	East South Central	GRAINS/CEREALS	R\$ 363,00	\$121.00
25	01/09/2013	East North Central	CONFECTIONS	R\$ 116,16	\$19.36
Total sales on the day - 01/09/2013				R\$ 642,51	

Fields Settings - General Total Visual setting

Sets the formatting of the fields displayed in the General total.

Visual setting of the General Total		
ATTRIBUTE	VALUE	DESCRIPTION
Text Font	<input type="text" value="Aa"/>	Field Text size
Font Size	<input type="text" value=""/>	Font Size
Font Color	<input type="text" value=""/>	Font Color
Background Color	<input type="text" value=""/>	Background Color
Bold text	<input type="text" value=""/>	Formating text in bold.

- **ont family** : Sets the font used.
- **ont size** : Sets the font size.
- **ext color** : Sets the text color
- **ackground color** : Sets the background color.
- **old** : Format text in bold.

Example of formatting the Grid General Total

Id	Date	Region	Product Category	Sales	Profit
1	01/01/2015	South Atlantic	SEAFOOD	R\$ 50,82	\$14.52
2	01/01/2016	South Atlantic	CONFECTIONS	R\$ 84,70	\$33.88
3	01/01/2013	South Atlantic	BEVERAGES	R\$ 90,75	\$36.30
4	01/02/2013	East South Central	CONFECTIONS	R\$ 163,35	\$54.45
5	01/02/2014	East South Central	DAIRY PRODUCTS	R\$ 251,68	\$94.38
Grand Summary				R\$ 267.620,19	

Settings

Summary General Settings.

Summary Settings

ATTRIBUTE	VALUE
Title	<input type="text"/>
Quantity Title	<input type="text" value="{lang_othr_rows}"/>
Horizontal Total	<input checked="" type="checkbox"/>
Vertical Total	<input checked="" type="checkbox"/>
The chart icon positioning	<input checked="" type="radio"/> left <input type="radio"/> right
Positioning of the Total icon	<input checked="" type="radio"/> left <input type="radio"/> right
Page Width	<input type="text" value="0"/>
Width unit	<input type="text" value="Automatic"/>
Display SubTotal label	<input type="text" value="Value"/>
Display line number	<input checked="" type="checkbox"/>
Display the hover on the Summary lines	<input checked="" type="checkbox"/>
Short value	<input type="checkbox"/>
Display icon only on mouseover	<input checked="" type="checkbox"/>
Fixed label	<input type="checkbox"/>

- **itle**: Defines a title for the Summary, if not informed the default title will be "Summary". For no title to be displayed, enter the HTML tag ` `.
- **uantity Title**: Title Amount of Record.
- **orizontal Total**: Show horizontal total for Summaries of matrix type.
- **ertical Total**: Display vertical total for Summaries of matrix type.
- **he chart icon positioning**: Chart icon positioning (left or right).
- **ositioning of the Total icon**: Total icon positioning (left or right).
- **age Width**: Width value for the Summary page.
- **idth unit**: Unit of measure used for width. Auto (width value is ignored), pixels and percentage.
- **isplay SubTotal label**: Displays the Total label or the Value itself.
- **isplay line number**: Display the sequence number of the record in the Summary.
- **isplay the hover on the Summary lines**: Apply the hover attribute by hovering the mouse cursor over the Summary lines.
- **hort value**: Displays the value abbreviated in the summary.
- **isplay icon only on mouse over**: Displays the sorting icon only when the mouse is on the field label.
- **ixed Label**: Establishes the column label in the screen top during the page scrolling.

Layout Settings

You can define the layout of the Summary for each type of Group By.

rule1	GROUP BY FIELD	POSITION	SORTING	FILL EMPTY LABELS	LINK GRID	ALIGNMENT
	date_YYYY	<input checked="" type="checkbox"/> Y axis	Database value	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="text"/>
	date_MM		Database value	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="text"/>

Tabular format

- **roup By Field**: Group By fields selected in the Grid.
- **osition**: Defines whether to use the x-axis or y-axis position.
- **orting**: Sets the sorting by the database value or by the display value.
- **ill Empty Labels**: Defines whether the empty labels will be filled.
- **ink Grid**: Creates a link in the selected field.
- **ignment**: Sets the layout alignment type to center, left, or right.
- **abular format**: Sets the Summary to the tabular format.

Toolbar

The toolbar is divided into two parts, top and bottom, so you can define which buttons will be displayed in both places. Button selection works independently.

You can also define which buttons will be displayed when the application is accessed by a mobile device, just access the Mobile tab.

For more information about the toolbar, access Application > Grid > Toolbar.

TOOLBAR

ATTRIBUTE	VALUE	DESCRIPTION
Configure the toolbar below for a "classic web version " and also for a "mobile version".		
Desktop Mobile		
Top Toolbar	Navigation Exit Export PDF WORD XLS XML CSV RTF Print Others Configure Detail Group By	Buttons organization on top toolbar. Left {lang_btms_expt} PDF ----- WORD XLS ----- XML CSV RTF ----- Print Chart Settings Configure
	Bottom Toolbar	Navigation Exit Export PDF WORD XLS XML CSV RTF Print Others Configure Detail Group By

Application Hotkeys

Scriptcase allows creating shortcut keys to your applications. You can select a predefined template or create specific actions for an application.

VALUE	DESCRIPTION				
<input checked="" type="checkbox"/> toggleHotkeys();	Define if the application will use hotkeys				
SC_DefaultHotkeys	Select the hotkey template from previously created schemas				
Clear <input type="button" value="+"/>					
<table border="1"> <thead> <tr> <th>ACTION</th> <th>KEYBINDING</th> </tr> </thead> <tbody> <tr> <td colspan="2" style="text-align: center;">No hotkeys configured</td> </tr> </tbody> </table>		ACTION	KEYBINDING	No hotkeys configured	
ACTION	KEYBINDING				
No hotkeys configured					
Clear <input type="button" value="+"/>					

Use hotkeys

Defines if the application uses hotkeys. When you enable this option, the default shortcut keys settings are disabled.

Hotkeys template

Select the [hotkey template](#) previously created.

Action

Selects the triggered action when pressing the selected key.

Keybinding

Selects the keys responsible for executing the chosen action.

Add "+"

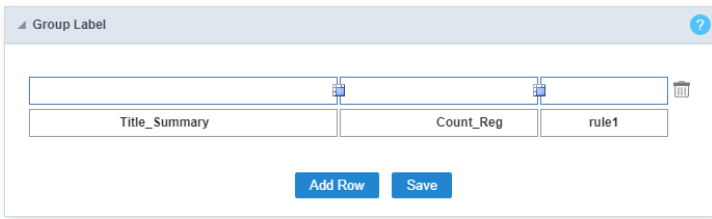
Adds a new action on the keys list.

Clear

It clears the selected hotkeys preference.

Group Label

Click on Add Row, so that the group label can be added.

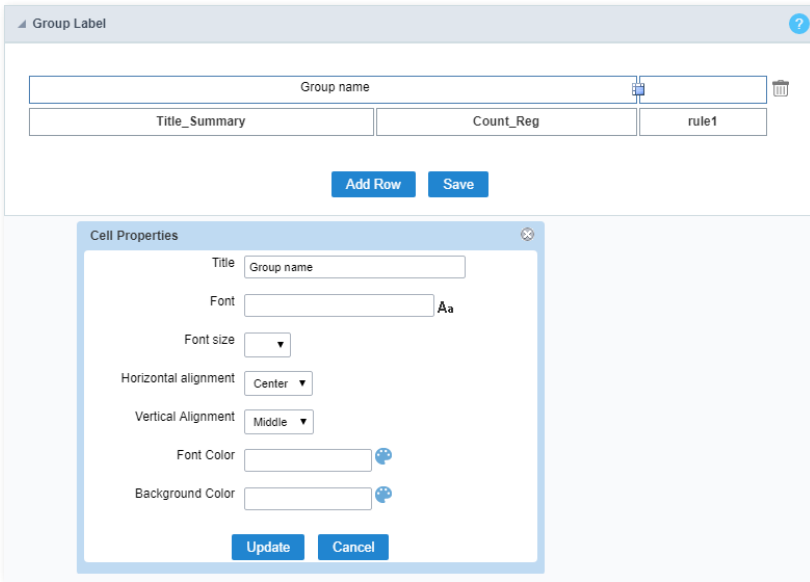


With this option, you can enter one or more titles for the columns of your applications.

If you have dynamic display fields, Group Label will not work.

To merge two or more cells, click the icon positioned between them.

To add a title, click the pencil icon to add a title.

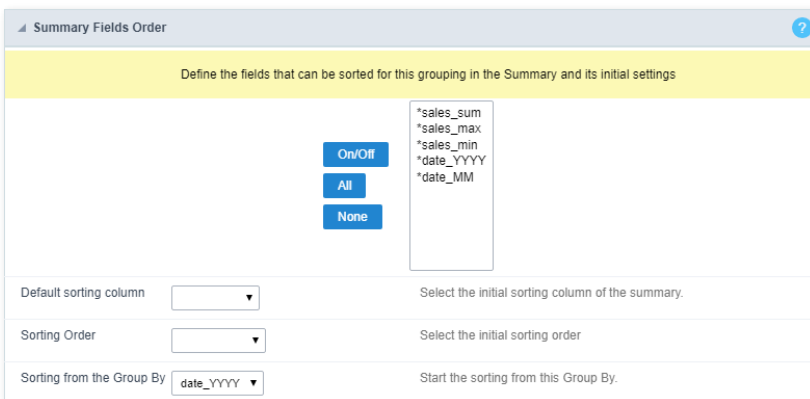


- Title** Sets the title that will be displayed in your Group Label.
- Font** Sets the font type of the title of your group label.
- Font Size** Sets the font size of the title of your group label.
- Horizontal alignment** Sets the horizontal alignment of the title of your group label. Being left, center or right.
- Vertical alignment** Sets the horizontal alignment of the title of your group label. Being middle, low or top.
- Font Color** Sets the color of the title text of your group label.
- Background Color** Sets the background color of the space where the title of your group label will be displayed.

Sorting

You can define the fields in which you wish to allow the sorting, when running the application.

This configuration must be performed in each of the application group by.



- **ields**: Select the fields that you want to allow sorting.
- **efault sorting column** : Select a field for initial sorting in the Summary.

- **orting Order:** Choose whether the sort will be ascending or descending.
- **orting from the Group By :** Select one of the Group By for initial sorting.

Limit

Lets you limit the amount of records to be retrieved from SQL and displayed in the Summary.

This option is only available for static group by.

Limit settings		
ATTRIBUTE	VALUE	DESCRIPTION
Field	id	Field to perform limit filter
Summarization	Sum	In which summarization function the limit will be applied
Limit type	Top	Set if the limit will be realized for the first or for the last.
Quantity		Number of records to be applied for the limit

- **ield:** Defines the field that will be used to perform the limit.
- **ummarization:** Defines the summarization function in which the limit will be performed.
- **imit type:** Sets the sorting that the limit will be applied, DESC or ASC.
- **uantity:** Sets the amount of records that will be returned.

Charts

One of the modules of the Grid application is the charts, which in turn are generated based on Grid summary information.

Scriptcase charts are generated in HTML5.

Chart Settings

In the settings screen, you can edit the settings of all available charts.

Chart Settings		
ATTRIBUTE	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> Bar	Font size	
<input checked="" type="checkbox"/> Pie	Abbreviated value	<input type="radio"/> Yes <input checked="" type="radio"/> No
<input checked="" type="checkbox"/> Line	Export Chart	<input type="radio"/> Yes <input checked="" type="radio"/> No
<input checked="" type="checkbox"/> Area	Subtitle	Below
<input type="checkbox"/> Gauge	Bar - Orientation	Vertical
<input type="checkbox"/> Radar	Bar - width (between 20 and 70)	
<input type="checkbox"/> Funnel	Bar - Value orientation	Vertical
<input type="checkbox"/> Pyramid	Bar - Value position	Inside
	Bar - Dimension	3D
	Bar - Stacking	off
	Bar - Series group	Set

To edit the specific settings for each Chart, select the Chart you want to edit.

To define the available charts when running the application, check the box next to the name of the Charts.

Below you will detail the specific settings of each of the Charts.

Bars

- **ont Size:** Sets the font size in the Chart. Enter only the font size, for example: 15.
- **bbreviated value:** Defines whether or not the values displayed in the Chart should be abbreviated.
- **ubtitled:** Sets the subtitled position of the Chart.
- **ar - Orientation:** Orientation of the vertical or horizontal bars on the Chart.
- **ar - width (between 20 and 70):** You can set the width from 20 to 70.
- **ar - Value orientation:** The orientation of the Chart is defined and can be placed vertically.
- **ar - Value position :** You can choose where the value will position itself.
- **ar - Dimension:** Bar Dimension (2D or 3D) on the chart.
- **ar - Stacking:** Stack the Bars in a single Bar (In series).
- **ar - Series group:** Group the Bar chart in series.

Pie

- **ont Size:** Sets the font size in the Chart. Enter only the font size, for example: 15.
- **bbreviated value:** Defines whether or not the values displayed in the Chart should be abbreviated.
- **ubtitled:** Sets the subtitled position of the Chart.
- **ie - Format:** Pie or Donut format.
- **ie - Dimension:** Dimension of the Pie Chart.
- **ie - Sorting:** Ordering the Pie Chart.
- **ie - Values format:** Formatting the displayed data.

Line

- **ont Size:** Sets the font size in the Chart. Enter only the font size, for example: 15.
- **bbreviated value:** Defines whether or not the values displayed in the Chart should be abbreviated.
- **ubtitled:** Sets the subtitled position of the Chart.
- **ine - Shape:** Line Format
- **ine - Series group:** Line grouping type.

Area

- **ont Size:** Sets the font size in the Chart. Enter only the font size, for example: 15.
- **bbreviated value:** Defines whether or not the values displayed in the Chart should be abbreviated.
- **ubtitled:** Sets the subtitled position of the Chart.
- **rea - Shape:** Format of the area displayed on the Chart.
- **rea - Stacking:** Stack the areas on the Chart.
- **rea - Series group:** Group the Chart in series.

Gauge

- **ont Size:** Sets the font size in the Chart. Enter only the font size, for example: 15.
- **bbreviated value:** Defines whether or not the values displayed in the Chart should be abbreviated.
- **ubtitled:** Sets the subtitled position of the Chart.
- **auge - Shape:** Display format of the Chart.

Radar

- **ont Size:** Sets the font size in the Chart. Enter only the font size, for example: 15.
- **bbreviated value:** Defines whether or not the values displayed in the Chart should be abbreviated.
- **ubtitled:** Sets the subtitled position of the Chart.

Funnel

- **ont Size:** Sets the font size in the Chart. Enter only the font size, for example: 15.
- **bbreviated value:** Defines whether or not the values displayed in the Chart should be abbreviated.
- **ubtitled:** Sets the subtitled position of the Chart.
- **unnel - Dimension:** Funnel dimension (2D or 3D) on the chart.

Pyramid

- **ont Size:** Sets the font size in the Chart. Enter only the font size, for example: 15.
- **bbreviated value:** Defines whether or not the values displayed in the Chart should be abbreviated.
- **ubtitled:** Sets the subtitled position of the Chart. **Pyramid - Dimension:** Dimension (2D or 3D) on the Chart. **Pyramid - Values format:** Formatting the displayed data. **Pyramid - Sliced:** Display format of the Chart.

Below are the settings common to all Chart types.

Summary chart type	<input type="text" value="Both - starting by synthetic"/>	Chart generation mode.
Create link on the chart	<input type="text" value="Create link to a child chart"/>	Creates link inside the chart to display the related data
Display Values	<input type="checkbox"/>	Display the values of the generated chart.
Axis of Chart General Total	<input type="text" value="Column"/>	Display grand total chart as bar chart or line chart.
Display Y-Axis With Zero.	<input checked="" type="checkbox"/>	Force zero display on the axis Y.
Label orientation on the x-axis	<input type="text" value="Horizontal"/>	Defines the labels orientation on the x-axis as vertical or horizontal. If you choose horizontal option and there are many values to be displayed, in such a way that they do not fit in the scale, the orientation will be automatically changed to vertical. This option is only available for charts type: Column, Line and Area.
Chart Width	<input type="text" value="800"/>	Chart width in pixels.
Chart Height	<input type="text" value="600"/>	Chart height in pixels.
Values sorting	<input type="text"/>	Sort values in charts.

- **ummary chart type:** Defines how the Charts will be generated: analytical, synthetic, or both.
 - **ynthetic** They should restrict themselves to the first Group By the condition and allow the user to use the links to see details of the next set of criteria, increasing the level of detail.
 - **nalytical** They appear with the whole Group By criteria, creating a complete detailed Chart that does not allow the linking of other detailed Charts.

- **reate link on the chart:** Allows the Charts to have a link in their elements for detailed Grid applications or Charts. The data shown will be relative to the value clicked on the Chart.
- **isplay Values :** Displays the values of the generated Chart.
- **xis of Chart General Total :** Option to display the Chart of the grand total as column or row.
- **isplay Y-Axis With Zero. :** Force display of the zero value on the Y axis.
- **abel orientation on the x-axis :** Sets the orientation of the labels on the X axis to vertical or horizontal. If the horizontal option is chosen and there are many values to be displayed so that they do not fit in the scale, the orientation will be automatically changed to vertical. This option is only available for Charts of type: Column, line and area.
- **hart Width:** Width in pixels of the generated Charts.
- **hart Height:** Height in pixels of the Charts generated.
- **alues sorting:** Sort the values of the Charts.

Column Charts

You can define the display of the column Charts as well as the label for the Chart and the axes. These settings are set individually for each Group By.

If the user sets up more than one field, the analytical chart will only generate with the first two.

Generate columns charts
?

Configure below which Charts will be displayed for each grouping and their basic parameters

Group by: **rule1**

	VIEW CHART	VIEW VALUE IN CAPTIONS	CHART TITLE	TITLE X	TITLE Y
date_YYYY					
sales - sum	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
sales - max	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
sales - min	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
date_MM					
sales - sum	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
sales - max	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
sales - min	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Line Charts (Totals)

The use of line Chart (totals) configuration is only available when you have a Group By with at least two fields.

To configure this option, go to the "Group By" menu and create a static Group By some fields or add fields to dynamic Group By.

Generate line charts (totals)
?

Configure below which Charts will be displayed for each grouping and their basic parameters

Group by: **rule1**

	VIEW CHART	VIEW VALUE IN CAPTIONS	CHART TITLE	TITLE X	TITLE Y
Grand Total					
sales - sum	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
sales - max	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
sales - min	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
date_YYYY					
sales - sum	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
sales - max	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
sales - min	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
date_MM					
sales - sum	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
sales - max	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
sales - min	<input type="checkbox"/>	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Layout

The Chart theme tool enables you to fully edit Chart themes.

These themes can be set only for the current application, when changed in the application itself or for all Charts when these settings are performed in the Charts editor.

For more information, access Layout > Chart Themes.

Fields

All fields added in the Summary totals, in the dynamic Group By or in the static Group By, will be listed here separated by the Group By name.

General Settings

General settings of total fields in the Summary.

General Settings: sales_sum		
ATTRIBUTE	VALUE	DESCRIPTION
Data Type	Integer	Field data type.
Type	Default	Field type.
Label	Sales ((lang_blns_smry_msge_sum))	Field title.
	Sales (Sum)	

- **ata Type**: Defines the data type for the field.
- **abel**: Defines the label for the displayed field in the Summary.

Values Formatting

Integer Type

General Settings: sales_sum		
ATTRIBUTE	VALUE	DESCRIPTION
Regional Settings	<input checked="" type="checkbox"/>	Apply regional settings to the field.
Color of Negative	<input type="text"/>	Color for negative values.

- **egional Settings**: When you activate it you apply the regional settings for this field. To configure them, click on the menu Locales > Regional Settings.
- **olor of Negative**: In this field you can enter a color in hexadecimal. (Example: # 000000)

Decimal Type

General Settings: sales_sum		
ATTRIBUTE	VALUE	DESCRIPTION
Regional Settings	<input checked="" type="checkbox"/>	Apply regional settings to the field.
Color of Negative	<input type="text"/>	Color for negative values.
Decimal precision	<input type="text" value="2"/>	Field decimal precision.
Complete with Zeros	<input checked="" type="checkbox"/>	Complete the field value with zeros.

- **egional Settings**: When you activate it you apply the regional settings for this field. To configure them, click on the menu Locales > Regional Settings.
- **olor of Negative**: In this field you can enter a color in hexadecimal. (Example: # 000000)
- **ecimal Precision**: Number of decimal places your field will have on display.
- **omplete with Zeros**: Enables the field value to be filled with zeros.

Currency Type

General Settings: sales_sum		
ATTRIBUTE	VALUE	DESCRIPTION
Regional Settings	<input checked="" type="checkbox"/>	Apply regional settings to the field.
Currency Format	<input type="checkbox"/>	Display field with currency format.
Color of Negative	<input type="text"/>	Color for negative values.
Decimal precision	<input type="text" value="2"/>	Field decimal precision.
Complete with Zeros	<input checked="" type="checkbox"/>	Complete the field value with zeros.

- **egional Settings**: When you activate it you apply the regional settings for this field. To configure them, click on the menu Locales > Regional Settings.
- **urrency Format**: When you enable this option the application will display the currency symbol according to the regional setting.
- **olor of Negative**: In this field you can enter a color in hexadecimal. (Example: # 000000)
- **ecimal Precision**: Number of decimal places your field will have on display.
- **omplete with Zeros**: Enables the field value to be filled with zeros.

Visual Settings of Totals in Group By

Visual settings of totalization in a Group By		
ATTRIBUTE	VALUE	DESCRIPTION
Text Font	<input type="text" value="Aa"/>	Field text font.
Font Size	<input type="text" value="v"/>	Font size.
Font Color	<input type="text" value="c"/>	Font color.
Background Color	<input type="text" value="c"/>	Background color.
Bold Text	<input type="text" value="v"/>	Bold text formatting.
Text Alignment	<input type="text" value="Align right"/>	Horizontal alignment of the text.
Vertical text alignment	<input type="text" value="Align on the top"/>	Vertical alignment of the text.

- **ont** : Allows you to define the font used in the Group By label.
- **ont Size** : Allows you to set the font size used in the Group By label.
- **ont Color** : Label color in Group By.
- **ackground Color** : Group By background color.
- **old Text** : Enables or disables the Group By label text in bold.
- **ext Alignment**: Sets the horizontal positioning of text.
- **ertical text alignment** : Sets the vertical positioning of text.

Visual Settings of The General Total

Visual setting of the General Total		
ATTRIBUTE	VALUE	DESCRIPTION
Text font	<input type="text" value="Aa"/>	Field text font.
Font Size	<input type="text" value="v"/>	Font size.
Font Color	<input type="text" value="c"/>	Font color.
Background Color	<input type="text" value="c"/>	Background color.
Bold Text	<input type="text" value="v"/>	Bold text formatting.
Text Alignment	<input type="text" value="Align right"/>	Horizontal alignment of the text.
Vertical text alignment	<input type="text" value="Align on the top"/>	Vertical alignment of the text.

- **ont** : Allows you to define the font used in the Group By label.
- **ont Size** : Allows you to set the font size used in the Group By label.
- **ont Color** : Label color in Group By.
- **ackground Color** : Group By background color.
- **old Text** : Enables or disables the Group By label text in bold.
- **ext Alignment**: Sets the horizontal positioning of text.
- **ertical text alignment** : Sets the vertical positioning of text.

Layout

Header


In this configuration, you can define the information that will be displayed in the header and footer of the Summary.

HEADER		
ATTRIBUTE	VALUE	
Display Header	<input checked="" type="checkbox"/>	
Insertion Title	<input type="text" value="{lang_othr_frmi_tit} - orders"/>	
Update Title	<input type="text" value="{lang_othr_frmu_tit} - orders"/>	
Header Variables		
SC_HEADER	<input type="text" value="Title"/>	
SC_VALUE	<input type="text" value="Date"/>	<input type="text" value=""/>

This screen may vary depending on the header format you choose within the Display folder.

- **isplay Header**: This flag determines whether the header will be displayed.
- **ummary Title**: Lets you enter a title to be displayed in the application.
- **ariables**: Variable fields can be filled with any of the types displayed in the Combobox, depending on the type you need to associate a content with the required one. Below you describe the existing types:
 - **ield**: When the "Field" option is chosen, it will open a Combobox next to the fields that are part of the "Select". By choosing one of these fields, you are associating the value of the field to display in the Header or Footer.
 - **ate** : When "Date" type is selected, the system date in mm/dd/yyyy format will be displayed in the Application Header or Footer. There are several display formats using the date



and time of the server. The format can be entered in the text field next to it. To access existing formats click on  and an explanatory window will appear.

- **mage** : When the image type field is selected, a field for filling in the image name on the server appears. To locate the existing images and select one, click on the “ **Choose Image**” icon and to make new images available on the server click on **“Upload”** .
- **alue**: When **“Value”** type is selected, the content that is filled in the text field next to option, appears in the Header or Footer. You can enter texts and **“ Global variables”**. Ex: **“Employee Name: [v_name]”**.

Footer

ATTRIBUTE	VALUE
Display Footer	<input type="checkbox"/>
Footer Variables	
NM_FOOTER_DATA	<input type="text"/>
NM_FOOTER_LOGO	<input type="text"/>
NM_FOOTER_TITLE	<input type="text"/>

This screen may vary depending on the Footer format you choose within the Display folder.

- **isplay Footer**: This flag determines whether the Footer will be displayed in the application.
- **ariables**: Variable fields can be filled with any of the types displayed in the Combobox, depending on the type you need to associate a content with the required one. Below you describe the existing types:
 - **ield**: When the **“Field”** option is chosen, it will open a Combobox next to the fields that are part of the **“Select”**. By choosing one of these fields, you are associating the value of the field to display in the Header or Footer.
 - **ate** : When **“Date”** type is selected, the system date in mm/dd/yyyy format will be displayed in the Application Header or Footer. There are several display formats using the date and time of the server. The format can be entered in the text field next to it. To access existing formats click on  and an explanatory window will appear.
 - **mage** : When the image type field is selected, a field for filling in the image name on the server appears. To locate the existing images and select one, click on the **“ Choose Image”** icon and to make new images available on the server click on **“Upload”** .
 - **alue**: When **“Value”** type is selected, the content that is filled in the text field next to option, appears in the Header or Footer. You can enter texts and **“ Global variables”**. Ex: **“Employee Name: [v_name]”**.

Search

Settings

ATTRIBUTE	VALUE	DESCRIPTION
Display combobox quantity	<input type="checkbox"/>	Displays the amount of existing records in the database next to the value in the filter combo.

- **isplay combo box quantity** : Allow to display the quantity of existence records in the table net to the value in search combo.

Settings Of Search Fields

To define the fields that will be used in the Summary search, use drag-and-drop to position the fields within Search Fields.

The order of the fields, within Search Fields, will be the display order in the generated application.

This search works only in Grid summary.

Search fields settings	
GRID FIELDS	SEARCH FIELDS
# id	T region_1 <input type="button" value="Edit"/> x
📅 date	# id_1 <input type="button" value="Edit"/> x
T region	\$ sales_1 <input type="button" value="Edit"/> x
T product_category	
T product	
T customer_name	
# qty sold	
\$ cost	
\$ sales	
\$ profit	

Editing Fields

The same field can be added more than once to the Search, these fields will be visible in the generated application since it uses distinct settings, if they are using the same configuration only one of the fields will be displayed.

The display settings of the fields are performed individually, and can be accessed by adding the field in the search and clicking edit.

you will separate the explanation of the display settings from the fields according to the types.

Types: Text/Special

Configuration interface for text and special fields.

Choose Component

You must define how the field will be used in the Search. Each field type, text, number, and date have different configuration options.

- **each field label** : Defines the label of the field that will be displayed in the Search.
- **choose component type** : It defines the format of use of the fields in the Search, for text fields you have the Select Box and the Multi select box.

Choose Values

The **default value** field allow us to define a default value to the summary search in the initial application charge.

Lookup

In this tab, you can configure the display lookup in the Search field. For more information on creating a view lookup, access **Application > Grid > Fields > Text**.

Types: Date/Time

Configuration interface for date and date time fields.

Search Settings x

Choose component
Choose values

Field label for display birthdate_1

Choose the display component:

Date Range
Use this option so that users can filter using a date range determined by them.

From 24 Apr 2017 00 00 00

To 15 Oct 2012 00 00 00

Actual Period Simple select ▼
Use this option so that users can filter according to an actual date period. For example: 2019, 2015, etc.

Relative Period Simple select ▼
Use relative period so that users can filter according to a period, for example: current year, last month, current quarter, etc.

Seasonal Period Simple select ▼
Use seasonal periods so that users can filter according to the repeated periods. For example: every month of April, every quarter, every Saturday, etc.

Ok
Cancel

Choose Component

You must define how the field will be used in the Search. Each field type, text, number, and date have different configuration options.

- **search field label** : Defines the label of the field that will be displayed in the Search.
- **hoose component type** : Defines the format of use of the fields in the Search, for fields date and datetime, you will have the following options: Date Range, Actual Period, Relative Period and Seasonal Period.

Choose Values - Date Range

For datetime fields, you must enable the option **Include Time** so that the hours can be included in the search

Select below the date ranges to allow system users to filter the Summary according to the periods displayed:

Include time
Example: (From Jan 1st 2006 15:50:20 to Jan 1st 2010 20:15:50)

Choose Values - Actual Period

You must define the period that will be used in the search. When running the application, you will have a select with the dates displayed according to the selected period.

Select below the date ranges to allow system users to filter the Summary according to the periods displayed:

Year

▼

All the distinct values in the column will be listed

Default value:

Sets a default value for filter on the initial load of the application.

Choose Values - Relative Period

You must define which periods are available for use in the search.

Select below the date ranges to allow system users to filter the Summary according to the periods displayed:

Select values

Year Quarter Month Week Day Time only

Current year

Last year

Next year

Default value:

None ▾

Sets a default value for filter on the initial load of the application.

Specify the default filter value to apply. Ensure that the value is present in the column. Enclose the comma containing values inside the double quotes.

Single: Last Year
Multi: Last Year,Current Year

Choosing Values - Seasonal Period

The available values are separated by tabs:

Select below the date ranges to allow system users to filter the Summary according to the periods displayed:

Select values

Quarter Month Week Week Day Day Time only

Quarter 1

Quarter 2

Quarter 3

Quarter 4

Default value:

None ▾

Sets a default value for filter on the initial load of the application.

Specify the default filter value to apply. Ensure that the value is present in the column. Enclose the comma containing values inside the double quotes.

Single: October
Multi: May, October

- **uarter:** Defines the quarters used in the search.
- **onth:** Defines the usage of the months of the year.
- **eeek:** Defines the usage of the weeks of the year.
- **eeek Day:** Defines the use of the days of the week.
- **ay:** Defines the usage of the days of the month.
- **ny Time:** Defines the use of the day time (Available only in the datetime field)

Types: Number

Configuration interface for numeric fields.

Search Settings

Choose component Choose values Lookup

Field label for display: id_1

Choose a function: Sum

Choose the display component:

Slider Range to be incremented

11 80

Use the slider component to allow the user to quickly filter numerical data by sliding a graphical thumb between the end points of a track that corresponds to a range of values.

Ok Cancel

Choose Component

You must define how the field will be used in the Search. Each field type, text, number, and date have different configuration options.

- **Search field label** : Defines the label of the field that will be displayed in the Search.
- **Choose function to apply** : Defines the function that will be used in the search for setting the value. For example, when choosing sum, the range is the range between the lowest and the result of the sum greater.
 - **Actual Values**: This option sets the display of the actual value of the field, saved in the database, without the use of any of the aggregate functions. When using this option, two types of searches, select box and Multi select box usage are added, as well as enabling lookup settings for numeric fields.
- **Choose component type**: Defines the format used for the fields in the search, for numeric fields, you have the Range and selecting **Actual Values** will be displayed **Select Box** and **Multi select box**.

Lookup

*For numeric fields, this option is available only when you use the **Actual Values** option and the way the field is used in the search is **Select box** or **Multi select box**.*

In this tab, you can create a display lookup in the search field. For more information about creating view lookup, go to **Application > Grid > Fields > Integer** .



1. [Home](#)
2. [Applications](#)
3. Grid
4. Settings

Settings

With this interface, you can define general options of the Search Form.

Search configuration Interface.

Search Criteria

Allows to select the logical operator **AND** or **OR** to define the criteria of the search;

Display Condition

Gets the condition of the search available for the user to choose one. He can select "AND" or "OR" in a Combobox.

Use auto-complete in the fields

Automatically turns the field into an autocomplete according to the existing values in the database. If the user chooses **Yes**, the autocomplete will enable automatically in all inputs that contain a relationship. If the user decides **No**, so no autocompletes will be displayed. Otherwise, the option selected is **Defined in the field** it'll keep the settings for each field individually.



1. [Home](#)
2. [Applications](#)
3. Grid
4. Search Criteria

Search Criteria

With this interface, you can configure the conditions available for each field of the Search form.

Search configuration Interface.

We can see the fields list on the left combo. On the right, the list of options for filtering the selected field. To select an option, click on one of them (Equal to, Beginning with, Contains, etc.) and then the button On/Off. The arrows, on the right, allows altering the order of the fields.

For the Date type fields, you can define special conditions for the search, accessing the field configurations, and editing the Special Conditions Settings.

Below the list are the buttons to enable the selected options:

- **On/Off:** Enables or disables the field or the option chosen.
- **All:** Marks all fields or options.
- **None:** Unmarks all the fields or options.



1. [Home](#)
2. [Applications](#)
3. Grid
4. Settings

Settings

Through the table below we set all the options that will be part of the application Grid Search.

Horizontal Alignment

Allows you to define the Horizontal positioning of the Search Form.

Margins

Defines the position of the margins of the Search Form.

Keep Values

It keeps the searched values when the user returns to the search form.

Keep Columns and Order Selection

Set it to preserve the selected columns and sorting for each search, if they went changed by the user through the toolbar options.

Use Enter to

It allows you to define the action that the Enter Key has on the Search form. **Tabulate** enables you to navigate between fields, and **Submit** performs the search (activates the Search button).

Display Tags

Allows displaying as tags, the searches used for the Grid.

Display after filtering

Display tags only after performing an advanced search. If disabled, it will always display a tag, regardless of the advanced search.

Unify results

Sets the chars limit to group the result of the tags. This option should be used when the field type is multiple-select.

Treeview in the Tags

Sets the use of Treeview for tags.

Initial status of the Treeview

It sets the initial state of the Treeview. (If the app is using Treeview for tags)

Start open

It displays the tags.

Start close

It displays the full description of the tags as text.



1. [Home](#)
2. [Applications](#)
3. Grid
4. Select Fields

Select Fields

![Interface for filter fields selection.][filtro_avancado_configuracao_selecionar_campos]



1. [Home](#)
2. [Applications](#)
3. Grid
4. Required

Required

Defines which fields of application will be required for the search.

![Required fields interface.][filtro_avancado_configuracao_campos_obrigatorios]

The application generated will be displayed a bullet (*) next to the field and an error message is generated if not assigned no value.

![Configuration interface of the marker placement.][filtro_avancado_configuracao_campos_obrigatorios_posicionamento]

- **Marker position** : Marker's position relative to the field.
- **Display message** : Displays whether or not the validation error message.

□

1. [Home](#)
2. [Applications](#)
3. Grid
4. Toobar

Toobar

Desktop

The Search toobar is divided in two parts: Top and Bottom, in a way that is possible to define the buttons that will be displayed in both bars. The selection of buttons in the top and bottom toobar works independently, allowing the buttons to be displayed in both bars at the same time.

Toobar Interface.

Navigation: Groups the options relative to the navigation buttons that can be displayed in the application.

- **Search:** Execute the search.
- **Clean:** Clean the all the search fields.
- **Edit:** Enable the **Save Tag** option.
- **Exit:** Exit the application.

Others: Groups a diversity of options relative to the application.

- **Languages:** Displays a combobox with the names available, defined in the project properties.
- **Themes:** Displays a combobox with the themes available, defined in the project properties.
- **HelpCase:** Displays a button to redirect to the help page.

Separator

- **-----:** Displays a line separating the buttons, when used the Group Buttons.

Use in-line buttons: Allows the alignment of the filter buttons next to the fields.

- **__ Inline buttons__:** Allows you to select which buttons will be displayed next to the field, and you can sort them according to your wishes. This option is available by enabling **Use Online Buttons** in the button settings.

Button Settings

Button Settings Interface.

- **Hotkey:** Allows you to set keyboard hotkeys to a button.
- **Use Inline Buttons:** Allows alignment of the filter buttons next to the fields.
- **Position of the in-line buttons:** Sets the positioning of the buttons to the right or left of the fields.
- **Column Quantity:** Sets the number of display columns of the buttons, allowing you to configure whether they will be displayed side-by-side or distributed in columns.

Options

Options Interface.

- **Button Position(Top/Bottom)** Positioning the buttons of the toobar Top/Bottom.



1. [Home](#)
2. [Applications](#)
3. Grid
4. Save Search

Save Search

This feature allows the end-user to save his searches in a profile. You can create some rules, like to save the searches by user login.

Save Filter Interface.

Save Filter Interface.



1. [Home](#)
2. Applications

General Overview

In events, blocks can be used with global variables, local, JavaScript code, CSS codes and Scriptcase macros.



1. [Home](#)
2. [Applications](#)
3. Grid
4. onScriptInit

onScriptInit

This event occurs when the application scripts are started. It runs before the OnLoad events(when present), however, it will always be executed when running, reloading or navigating the application.



1. [Home](#)
2. [Applications](#)
3. Grid
4. onRefresh

onRefresh

This event runs when the refresh option of the application is enabled.



1. [Home](#)
2. [Applications](#)
3. Grid
4. onSave

onSave

This event runs when the application saves the record.



1. [Home](#)
2. [Applications](#)
3. Grid
4. onValidade

onValidade

This event runs when validating the information of the fields, when submitting the form.



1. [Home](#)
2. Applications

General Overview

On this module are available the features of editing, attributes and display of the application, in a way that you can apply display themes, organize blocks, define values and the format of the Header/Footer and etc.

1. [Home](#)
2. [Applications](#)
3. Grid
4. Blocks

Blocks

Conceptually a block is a “container” where you can position the fields of the Applications: Form, Control or Grid with Slide orientation.

By default, the applications created in ScriptCase are built with a only one block, with the same name as the application. You can add the amount of blocks that you want to organize in a more convenient way. The page below, observe that theirs a column Organization, and that is where you'll define if the next block will be set beside or below the current one.

Application Block configuration

On the left side of each block there are two icons, first has the function to edit all the information relative to the block and the second is to delete the block.

- Organizing the position of the Blocks
 - See below how to modify the display order of the Blocks in one Page.

Click and drag the block that you desire to modify to its new position.

Application Block Display configuration

- See how to remove a block from display

Click on the block desired and drag it to the item “Blocks not Shown”. This way, you can also drag the block to another page if desired. See the images below.

Application Block Display configuration

Application Block Display configuration

- **Attributes**
 - **Block**
 - **Name** : Name of the Block.
 - **Label** : Title of the block that'll be displayed in the application.
 - **Title**
 - **Display** : Flag that controls the title display of the block.
 - **Label**
 - **Display** : Flag that controls id the label of the fields will be displayed in the block.
 - **Position** : Options to display label :
 - **Above** : The label will be displayed above the field.
 - **Beside** : The label will be displayed beside the field.
 - **Below** : The label will be displayed below the field.
 - **Fields**
 - **Columns** : Amount of columns that are displayed side by side in the block.
 - **Position** : The way that the fields are displayed in the block :
 - **Below** : The fields are displayed one below the other respecting the amount of columns.
 - **Beside** : The fields are displayed one beside the other respecting the amount of columns.
 - **Line** : The fields are displayed one beside the other without the tabulation.
 - **Organization**
 - **Next** : The way that the blocks are displayed in the page:
 - **Below** : Indicates that the next block will be placed below the current one.
 - **Beside** : Indicates that the next block will be placed beside the current one.
 - **Tabs** : Indicates that the next block will be placed in a different tab then the current one.
 - **Width** : Specifies the width that block will occupy in pixels or percentage, in case the value is in percentage, inform the (%).
 - **Collapse** : Enables the option to close the block.
- Create New Block

To include new blocks in an Application, click on the button . Next, you'll see the following interface to define the name and label of the block. At the end click on Create.

Creating application blocks configuration

- **Attributes**
 - **Name** : Name of the Block.
 - **Label** : Title of the block that'll be displayed in the application.
- Edit Blocks

To edit a block just click on the icon , that is on the left side of the block. Next, you'll see the following interface to define the parameters of the blocks. At the end click on save.

- **Name** : Name of the block.
- **Title** : Block title for display.
- **Display Title** : This option, when active, allows to display the block title.
- **Title Font** : Font applied to the block title.
- **Font Size** : Size of the font applied to the block title.
- **Font Color** : Font color for the block title.
- **Background Color** : Background Color of the block title.
- **Background image** : Background image for the block title.
- **Title Height** : Height in pixels of the block title line.
- **Horizontal Alignment** : Horizontal Alignment of the block title (Left, Center and Right).
- **Vertical Alignment** : Vertical Alignment the block title (Top, Middle and Bottom).
- **Display Label** : Display the labels of the fields in the block.
- **Columns** : Amount of field columns in a block.
- **Columns Width** : How the width of the block is defined.
- **Label Color** : Color of the field labels.
- **Fields Organization** : How the fields are organized in the block.
- **Label Position** : Position of the field labels relating to the data.
- **Next Block** : Position of the next block relating to the current block.

- **Border Color** : Border color for the block.
- **Border Width** : Border Width for the block.
- **Block Width** : Width for the block.
- **Block Height** : Height for the block.
- **Cell Spacing** : Cell Spacing in the block.
- **Collapse** : Enables the option to close the block.



1. [Home](#)
2. [Applications](#)
3. Grid
4. Settings

Settings

On this interface, you can define the theme for display of a specific application, this being because ScriptCase uses a Standard Definition of Values per project, that besides the Display Theme, allows to define values for other attributes on a Project level.

Header Template:

Allows to choose the template that's going to be used as the Header of the application.

Footer Template:

Allows to choose the template that's going to be used as the Footer of the application.

Button:

Allows to choose the button theme for the application.

Themes:

Choose one of the existing themes, it'll load the display mode (colors ,fonts, etc.) that'll be part of the application.

1. [Home](#)
2. [Applications](#)
3. Grid
4. Header & Footer

Header & Footer

Header

In this block, it's the definition of the variables content that'll be part of the header.

This page may change depending on the header format chosen in the Layout Settings.

Display Header:

This option determines if the header will display.

Title:

Allows to inform the title displayed in the application.

Header Variables:

The field variables can be informed with anyone of displayed in the Combo box. Depending on the type, it'll be necessary to associate the content with the field. Below there are the types of content: * **Field** : When you choose the option **'Field'**, it'll open a Combo box beside to choose the field you want. Selecting the desired field, it'll associate the value of the field with the header. * **Title** : This option when selected it'll display in the header the value informed in the **Application Title**. * **Date** : When selected the **'Date'** type, it'll display the system's date in the header. There are a diversity of formats using the date and time of the server. The format can be informed in the text field that appears beside the field. To access the existing formats, click on the icon and you'll view a page display the formats. * **Image** : When selecting the image type, it displays a field to inform the name of the existing image in the server. To locate the images existing and selecting one, click the icon **"Choose Image"** and to upload new images click on the button **'Upload'**

* **Value** : When selecting the type **'Value'**, the content informed in the text field that appears beside, it'll be displayed in the header, you can inform texts and **"Global Variables"**. e.g. "Employee Name: [v_name]".

Depending on the Application, you may have more than one title option.

Footer

This page may change depending on the footer format chosen in the Layout Settings.

Display Footer:

This option determines if the footer will display.

Footer Variables

The field variables can be informed with anyone of displayed in the Combo box. Depending on the type, it'll be necessary to associate the content with the field. Below there are the types of content: * **Field** : When you choose the option **'Field'**, it'll open a Combo box beside to choose the field you want. Selecting the desired field, it'll associate the value of the field with the footer. * **Date** : When selected the **'Date'** type, it'll display the system's date in the footer. There are a diversity of formats using the date and time of the server. The format can be informed in the text field that appears beside the field. To access the existing formats, click on the icon and you'll view a page display the formats. * **Image** : When selecting the image type, it displays a field to inform the name of the existing image in the server. To locate the images existing and selecting one, click the icon **"Choose Image"** and to upload new images click on the button **"Upload"**. * **Value** : When selecting the type **'Value'**, the content informed in the text field that appears beside, it'll be displayed in the footer, you can inform texts and **"Global Variables"**. e.g. "Employee Name: [v_name]".



1. [Home](#)
2. [Applications](#)
3. Grid
4. Settings

Settings

Dynamic Search configuration Interface

- **Use the ENTER key to** : Use the Enter key to tabulate from one field to the other, or to submit the search.



1. [Home](#)
2. [Applications](#)
3. Grid
4. Select Fields

Select Fields

Dynamic Search Field Selection.



1. [Home](#)
2. [Applications](#)
3. Grid
4. Settings

Settings

Quick Search is an option that allows to search data in various fields of the application, by using the text box in the toolbar.

Application running QuickSearch.

Quick Search Settings

In the quick search settings are the following options:

QuickSearch Configuration Interface.


Button within the search - An option to inform if the search button will be in the text area of the box. **Quick search show combo box** - Displays a combo box if there is only one option in the quick search. **Quick search Watermark** - Displays a Placeholder in the quick search. **Quick search width** - Defines the width of the Quick search field. **Display the Quick search old format** - Displays in the old format with the selection of the fields in the Quick search. **Search anywhere** - If enabled, QuickSearch will search each part of the String for the data informed in the field.




1. [Home](#)
2. [Applications](#)
3. Grid
4. Select Fields

Select Fields

You need to define the fields that are part of the search in the Quick search Settings .

 *QuickSearch selecting fields Interface.*

And you can select various criteria of the search.

 *QuickSearch search criteria configuration Interface.*

You need to add the QuickSearch button in the toolbar in order to use it.

1. [Home](#)
2. Applications

General Overview

The applications can have two types of fields.

Table Fields

Columns of the table connected to the application and their data types.

Example:

In a table with the following structure:

```
CREATE TABLE `orders` (
  `orderid` int(11) NOT NULL,
  `customerid` varchar(5) NULL DEFAULT NULL,
  `employeeid` int(11) NULL DEFAULT NULL,
  `orderdate` date NULL DEFAULT NULL,
  `requireddate` date NULL DEFAULT NULL,
  PRIMARY KEY (`orderid`) USING BTREE
);
```

These are the listed fields on left menu **Fields**:

Virtual Fields

Fields created inside application to assist in the development process.

How to create a virtual field

1. Inside an application, access the left menu **Fields** and click on **New Field** option.
2. Choose how many fields do you want to create:

1. Create the fields and define type, name and label for each one:

- **Type:** Data type of the field.
- **Name:** Internal name of the field. Used to identify the field on events and Scriptcase interface.
- **Label:** Title of the field displayed on the application.

Scriptcase **doesn't insert** virtual fields on the table after its creation.

Types of fields by applications

Application Table Fields Virtual Fields

Grid	<input type="text"/>	<input type="text"/>
Procedure	<input type="text"/>	<input type="text"/>
Chart	<input type="text"/>	<input type="text"/>
Form	<input type="text"/>	<input type="text"/>
Control	<input type="text"/>	<input type="text"/>
Search	<input type="text"/>	<input type="text"/>
PDF Report	<input type="text"/>	<input type="text"/>
Calendar	<input type="text"/>	<input type="text"/>

The applications **Menu**, **Tree Menu**, **Dashboard** and **Blank** were not listed because they haven't manage fields.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Text

Text

General Settings

This type of field allows the developer to create quickly fields to display data from the database, where the final user can see the data in the way it was set by the developer.

Type of Data

Defines the type of the field in the application. In this case we should select **Text**.

Label

Defines the title that will be displayed in the field when the application is executed. The terminology used in the interface is fundamental to the system have a good usability, when should use names and familiar terms to the final user of the application, instead of using terms from the system.

For example, this text field that has the name **customerid**, the client would have a much better understanding of the functionality of the field when we define the label as **Customer Name**.

Not only a fixed text, the **Label** attribute allows the use of langs to define the field's title, making it possible to use your application in a multi language project.

Case Settings

This option allows the developer to inform how the text will be displayed in runtime.

The options available are:

Lower Case: Every letter in the text will be converted to lower case.

Upper Case: Every letter of the text will be converted to upper case.

Capitalize first word: The first letter of the first world will be converted to upper case.

Capitalize all words: The first letter of every words will be converted to upper case.

Show HTML Content

When this option is active every HTML, CSS and JavaScript content that are in the database will be displayed with the main value.

Field Mask

Defines the field mask. There are two types of mask described in the table below:

Character Description

- X It is overridden by any characters returned by the database. Filled is required and the value will be completed with leading zeros when there are fewer characters than the size of the mask.
- Z It is overridden by any characters returned by the database. Its filling is optional and when there are fewer characters than the mask size, nothing will be done about the characters that are missing. In addition the leading zeros will be suppressed. When used in conjunction with the X necessarily the use of this character is to the left of the mask.
- 9 It represents any numeric character (from 0-9)
- A It represents an alpha numeric character (A-Z,a-z)
- * It represents any alpha-numeric character (A-Z,a-z,0-9)entered by the user.

For example, it is possible to set the mast to display a telephone number:

It will be show with this format on runtime:

It is also possible to set the field mask like those examples:

Field mask examples:

Telephone number

Field	Mask Input	Typed Value	Formatted Value
Phone Number	+99 99 9999 - 9999	123456789012	+12 34 5678 - 9012
Phone Number	(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
Phone Number	(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
Phone Number	(zz) xxxx - xxxx	12345678	() 1234 - 5678
Software Key (Letters only)	AAAA-AAAA-AAAA-AAAA	QWERTYUIASDFGHJK	QWER-TYUI-ASDF-GHJK
Software Key (Letters and numbers)	xxxx-xxxx-xxxx-xxxx	Q1W2E3R4T5Y6U7I8	Q1W2-E3R4-T5Y6-U7I8
License Plate	AAA - 9999	QWE1234	QWE - 1234
Scriptcase Serial Key	A999A999A999-**	D111H111G111DG2P	D111H111G111-DG2P
Multiple masks (Phone Number)	9999-9999;(99)9999-9999; 9999 999 9999		+99 99 9999-9999

Repeat Value:

When this option is active the value of the field will be repeated if the previous database register is the same.

Example:

SQL Type

Informs the type of the field in the database.

Field Behavior

Use autocomplete:

Field automatically turns into autocomplete according to existing values in the database.

Use Select2

Uses the new component for data selection, allowing searches within the select comboBox.

Width for the Select2

Sets a width for the area in the Select2.

Amount of characters

Sets the amount of characters to start the search.

Amount of rows

Sets the maximum number of rows to list the search result.

Width:

Defines a width in pixels for a result box.

Search options:

Defines the validation that will be made to fetch the search result.

Start equals to: Will return the records with the same start value as in the database.

Any part: Will return the records when exist the character in any part of the record.

End equals to: Will return the records with the same final value as in the database.

Position between values:

Defines the position that objects will be displayed.

Text Between Values

Text that will appear when using a filter condition between two values.

OnChange Submit:

Submit search on this field changing.

Show Condition:

To show or not the search condition. It only works if the search has at least one condition.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

• CSS of the Title

- **Font** : Allows to choose the font type, that will be applied to the application field title.
- **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
- **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.

• CSS of the Field

- **Font** : Allows to choose the font type, that will be applied to the application field.
- **Font Size** : Allows to choose the the font size, that will be applied to the application field.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
- **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the field.
- **Height** : To define a height for the field.

• CSS of the Input Object

- **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
- **Font Size** : Allows to choose the the font size, that will be applied to the application field.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.

- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the Input Object.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
- **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the Input Object.
- **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

1. [Home](#)
2. [Applications](#)
3. Grid
4. Integer

Integer

On this page, you will learn how you can configure settings related to the Number field. From the use of specific symbols display to the mode in which they are displayed. And thus, boost the application.

Data type:

It sets the application field type. When the field is set as Number, it is permissible to define formatting rules of integers.

Label:

It sets the title that will be displayed in the field when you run the application. The terminology used in the interface is fundamental for your system has a good usability. In this case, we recommend you to use names and terms familiar to the end user of the application, instead of using terms originated in the system.

Field Mask Input:

It sets the mask input for the field. There are two types of mask as described in the table below:

Character Description

X	It is overridden by any characters returned by the database. Filled is required and the value will be completed with leading zeros when there are fewer characters than the size of the mask.
Z	It is overridden by any characters returned by the database. Its filling is optional and when there are fewer characters than the mask size, nothing will be done about the characters that are missing. In addition the leading zeros will be suppressed. When used in conjunction with the X necessarily the use of this character is to the left of the mask.
9	It represents any numeric character (from 0-9)
A	It represents an alpha numeric character (A-Z,a-z)
*	It represents any alpha-numeric character (A-Z,a-z,0-9)entered by the user.

Mask Input Examples:

Field	Mask Input	Typed Value	Formatted Value
Phone Number	+99 99 9999 - 9999	123456789012	+12 34 5678 - 9012
Phone Number	(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
Phone Number	(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
Phone Number	(zz) xxxx - xxxx	12345678	() 1234 - 5678
Software Key (Letters only)	AAAA-AAAA-AAAA-AAAA	QWERTYUIASDFGHJK	QWER-TYUI-ASDF-GHJK
Software Key (Letters and numbers)	xxxx-xxxx-xxxx-xxxx	Q1W2E3R4T5Y6U7I8	Q1W2-E3R4-T5Y6-U7I8
License Plate	AAA - 9999	QWE1234	QWE - 1234
Scriptcase Serial Key	A999A999A999-**	D111H111G111DG2P	D111H111G111-DG2P
Multiple masks (Phone Number)	9999-9999;(99)9999-9999; 9999 999 9999		+99 99 9999-9999

Repeat value:

This option when enabled will allows you to repeat the field value if it is equal to the value of the previous record in the database.

Example:

Tipo SQL:

Reports the SQL type of the field as it is configured in the database.

Use regional settings:

It allows you to apply regional settings to the number fields formatting. When this option is not selected it will be displayed the attributes group, minus sign and negative number format.

Grouping:

It allows you to define which character is used to separate thousands. This option is only available when the option to use regional settings is disabled.

Negative sign:

It allows you to define which character will be used for the display of negative numbers. This option is only available when the option to use regional settings is disabled.

Negative number format:

It allows you to define the placement of the negative sign in relation to value. This option is only available when the option to use regional settings is disabled.

Color for the negative values:

It allows you to define a color when the value is negative, improving the understanding of the end user about that kind of value.

Example:

Display the value in words:

The value of the field will be displayed in full on application. This feature can facilitate the comprehension and understanding of the user.

Example:

Line size:

Maximum size in characters to be displayed in the value cell, in full. When this value is exceeded the line will break within the cell.

Lookup Settings

The Lookup is a way used to provide the end user a list of values description/name where the items can be selected and used in the application. For example, for a field sex that in the database is represented with M or F values can be presented as male or female dynamically with the lookup. So you will be able to retrieve these values from a database table or manually, if the reference table does not exist.

Example:

Lookup Methods

There are two lookup methods available:

Manual: To use this method it is necessary to inform the selection conditions manually to the Lookup.

Automatic: To use this method it is necessary to use a lookup query from the database to access the reference table.

Editing-automatic lookup:

Select Command:

It sets the SQL command that will retrieve the value to display inside the Grid field. The command must have the following format

```
SELECT field_to_be_displayed FROM table WHERE key_field = {grid_field}
```

The Grid field must be always referenced between curly braces {}, So at run time, the field braces will be replaced by its value in the table.

Multiple Options:

When the Select command informed return multiple values, the option must be selected.

Delimiter:

It sets the tab of values returned by the Select command, this option should be filled when the "Yes" option is checked in the multiple attribute options.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

Use a specific connection:

This option allows you to select another existing connection in the project, other than the current connection, to select the values for the lookup.

Edit Lookup - Manual:

Lookup used to modify the display of the field with predefined values.

Lookup Types

It sets the operating mode of the lookup can be: single value, multiple values (delimiter), multiple values (position) and multiple values (binary).

Single value: it will be displayed only one value for the field. There is a direct replacement of the value stored in the Bank for a label defined in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value of "M" will be replaced by "Male".

Multiple Values (Delimiter):

By using this option it will be displayed multiple values for the selected field. The values must be separated by a delimiter to be informed. A replacement of parts from a string stored in the field, separated by a delimiter for values contained in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value S;M will be replaced by Sports and Movies.

Delimiter: Character used to separate the values within the same string.

Multiple Values (Position)

Allows you to recover information stored in a single string of the selected field. In order for this information to be retrieved must be informed, in addition to the label, the starting position and the number of bytes that each information occupies within the string.

As an example we will use a string to store Sex, Marital status and Hobby respectively. Sex occupies one byte, Status one byte and Hobby two bytes .

For this purpose we define the list as:

Label	Value	Start	Size
Male	M	1	1
Female	F	1	1
Married	M	2	1
Single	S	2	1
Study	ST	3	2
Sportes	SP	3	2
Reading	LE	3	2

Example: the string MMST would be presented in the query as: Male Married Study

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value M will be replaced by Male.

Start: Starting position in the string where the information is recorded. The first position is always 1.

Size: Number of bytes that the information occupies in the string.

Multiple values (binary):

It allows you to retrieve several information stored in decimal form of the selected field.

As examples we will use the following list (although not informed in the interface of inclusion of values in the list, each item has a value assigned automatically according to the order):

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

To display the data, a decomposition of the decimal number stored in the bank is performed. For example, numbers 11 and 12 (written in the database) would be decomposed as follows:

$$11 = 1 + 2 + 8 = (\text{Sports} - \text{Culture} - \text{Reading})$$

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

$$12 = 4 + 8 = (\text{Leisure} - \text{Reading})$$

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

Label:

Text that will be displayed in the Grid.

Include Button:

Includes in the list the values populated in the Label and value fields.

Alter Button:

It changes the attributes of the selected item.

Delete Button:

Deletes the selected item.

Clean Button:

It cleans the fields

Save Button:

Allows the user to save the entire list, for later use using the Load Lookup definition option.

Load Lookup definition:

Allows you to load a predefined value list for use as a lookup.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

• CSS of the Title

- **Font** : Allows to choose the font type, that will be applied to the application field title.
- **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
- **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.

• CSS of the Field

- **Font** : Allows to choose the font type, that will be applied to the application field.
- **Font Size** : Allows to choose the the font size, that will be applied to the application field.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
- **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the field.
- **Height** : To define a height for the field.

• CSS of the Input Object

- **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
- **Font Size** : Allows to choose the the font size, that will be applied to the application field.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the Input Object.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
- **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the Input Object.
- **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Decimal

Decimal

General Settings

Decimal field Configuration Interface.

- **Data Type** : Select the type of field for the application. When it is defined as a Decimal you can define the format of a decimal number.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Accumulated** : When enabled, define that the field will be accumulated with the values of another field. This option is only available on virtual fields.
- **Accumulator field** : Only available when the option "Accumulated" is enabled. This configuration defines which field will be used to accumulate the values. The chosen fields needs to be Integer, Currency or Decimal.
- **Grid Mask** : Defines the mask for the field display. There are two typed of masks described below:

Character	Description
X	Placeholder to any character. Replaced by any character. If number of characters entered are less then the mask size, the field value is completed with zeros (Filling full size field entry is required).
Z	Replaced by any character retrieved from database. Suppress zeros at field left (Complete field filling is optional). When used combined with the mask character X it should be placed at the mask left.

Mask Examples

Mask	Field Value	Formatted Value
(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
(zz) xxxx - xxxx	1234567890	(12) 3456 - 7890
(z) xxxx - xxxx	12345678	() 1234 - 5678
(zz) xxxx - xxxx	0012345678	() 1234 - 5678

- **Do Not Repeat Value** : Does not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Field Behavior

Use autocomplete:

Field automatically turns into autocomplete according to existing values in the database.

Use Select2

Uses the new component for data selection, allowing searches within the select comboBox.

Width for the Select2

Sets a width for the area in the Select2.

Amount of characters

Sets the amount of characters to start the search.

Amount of rows

Sets the maximum number of rows to list the search result.

Width:

Defines a width in pixels for a result box.

Search options:

Defines the validation that will be made to fetch the search result.

Start equals to: Will return the records with the same start value as in the database.

Any part: Will return the records when exist the character in any part of the record.

End equals to: Will return the records with the same final value as in the database.

Position between values:

Defines the position that objects will be displayed.

Text Between Values

Text that will appear when using a filter condition between two values.

OnChange Submit:

Submit search on this field changing.

Show Condition:

To show or not the search condition. It only works if the search has at least one condition.

Values format

Decimal Field Format of Values with Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the number of the field. When not enabled, the attributes Digit Grouping Separator, Decimal Separator, Negative sign and Negative number format.
- **Decimal Precision** : Defines the amount of decimal places for the field.
- **Digit grouping** : Defines if the field will display the digits separator.
- **Decimal Separator** : Defines the decimal separator.
- **Accept** : Determines if the field, will accept only negative, positive or both numbers.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Currency

Currency

General Settings

![[Currency field Configuration Interface.][filtro_cons_moeda] *Currency field Configuration Interface.*

- **Data Type** : Define the type of field for the application. When set to Currency, you can currency values to the field.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Field Behavior

![[Currency Field Behavior Interface of the Search Configuration.][filtro_cons_moeda_filtro] *Currency Field Behavior Interface of the Search Configuration.*

- **Use auto-complete** : The field behaves as an auto-complete according to the values existing in the database.
- **Amount of characters** : Sets the amount of characters to start the search.
- **Amount of rows** : Sets the maximum number of rows to list the search result.
- **Width** : Sets the width in pixels for the result box.
- **Search options** : Defines the validation that will be made to fetch the search result.
- **Position between values** : This option sets the position that objects will be displayed.
- **Text between values** : Text that will appear between values.
- **OnChange Submit** : When enabled, it submits the search when there is modifications to the field.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Values format

![[Currency Field Format of Values with Regional Settings.][filtro_cons_moeda_format] *Currency Field Format of Values with Regional Settings.*

- **Regional Settings** : Allows to apply the Regional Settings to format the number of the field. When not enabled, the attributes Digit Grouping Separator, Decimal Separator.
- **Decimal Precision** : Defines the amount of decimal places for the field.
- **Digit grouping** : Defines if the field will display the digits separator.
- **Decimal Separator** : Defines the decimal separator.
- **Accept** : Determines if the field, will accept only negative, positive or both numbers.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Date

Date

General Settings

- **Data Type** : Define the type of field for the application. When set to Date, you can inform a date.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Watermark** : Displays a watermark in the field input.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the datatype of field in the database.

Values format

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator attribute.
- **Date separator** : Allows you to inform the separator symbol for the date.
- **Display** : Select the format of the day for display.
- **Internal Format** : Allows to define the format the field when the SQL type is different from **Date**. You need to use the characters **A**, **M** and **D** that correspond to **Year**, **Month** and **Day**.
- **Use Combo-box** : Allows you to select the date using a combo-box.
 - **Year as Combo** : Allows to use the year combo to select the date.
 - **Initial Year** : First year displayed in the combo.
 - **Actual Year +** : Display the current plus the amount of years informed.
- **Month in full textual** : Displays the Month format in Full.
- **Display Calendar** : Enables the a calendar icon beside the field, this allows to select the date from a calendar with the format already setup.
 - **New Calendar** : Defines if the JQuery calendar (New Calendar) is going to be displayed or the old format.
 - **Years Limit** : Amount of years displayed in the calendar.
 - **View week number** : Displays the number of the week in the application.
 - **Additional months** : Displays the additional months of the calendar.
 - **Show Combo year and month** : Displays the year and month of the calendar in the combo box.

Field Behavior

- **Position between values** : This option sets the position that objects will be displayed.
- **Text between values** : Text that will appear between values.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Special Conditions

- **All Period** : Searches for all periods of dates.
- **Today** : Searches in today's date.
- **Yesterday** : Searches in yesterday's date.
- **Last 7 days** : Searches the last 7 days. Ex: ((01/01/2017 01/07/2017)).
- **This month** : Searches the dates from the first day of the current month.
- **Last month** : Searches the dates from the first day of lasts month.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).

- **Width** : To define a width for the field.
- **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Grid
4. Time

Time

General Settings

![[Time field Configuration Interface.]][[filtro_cons_hora] *Time field Configuration Interface.*

- **Data Type** : Define the type of field for the application. When set to Time, you can inform a time to this field.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Values format

![[Time Field Format of Values with Regional Settings.]][[filtro_cons_hora_filtro] *Time Field Format of Values with Regional Settings.*

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator and time separator attributes.
- **Time separator** : Allows you to inform the separator symbol for the time.
- **Display** : Select the format of the time for display.
- **Internal Format** : Allows to define the format the field when the SQL type is different from**TIME**. You need to use the characters **HH**, **II**, and **SS** that correspond to **Day**, **Hour**, **Minutes** and **Seconds**.

Field Behavior

![[Time Field Behavior Interface of the Search Configuration.]][[filtro_cons_hora_format] *Time Field Behavior Interface of the Search Configuration.*

- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

1. [Home](#)
2. [Applications](#)
3. Grid
4. Datetime

Datetime

General Settings

![[Datetime field Configuration Interface.]][filtro_cons_data_hora] *Datetime field Configuration Interface.*

- **Data Type** : Define the type of field for the application. When set to Datetime, you can inform a date and time to this field.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Values format

![[Datetime Field Format of Values with Regional Settings.]][filtro_cons_data_hora_format] *Datetime Field Format of Values with Regional Settings.*

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator and time separator attributes.
- **Date separator** : Allows you to inform the separator symbol for the date.
- **Time separator** : Allows you to inform the separator symbol for the time.
- **Display** : Select the format of the day/time for display.
- **Internal Format** : Allows to define the format the field when the SQL type is different from **DATETIME**. You need to use the characters **A, M, D, HH, II, and SS** that correspond to **Year, Month, Day, Hour, Minutes and Seconds**.
- **Use Combo-box** : Allows you to select the date using a combo-box.
 - **Year as Combo** : Allows to use the year combo to select the date.
 - **Initial Year** : First year displayed in the combo.
 - **Actual Year +** : Display the current plus the amount of years informed.
- **Month in full textual** : Displays the Month format in Full.
- **Display Calendar** : Enables the a calendar icon beside the field, this allows to select the date from a calendar with the format already setup.
 - **New Calendar** : Defines if the JQuery calendar (New Calendar) is going to be displayed or the old format.
 - **Years Limit** : Amount of years displayed in the calendar.
 - **View week number** : Displays the number of the week in the application.
 - **Additional months** : Displays the additional months of the calendar.
 - **Show Combo year and month** : Displays the year and month of the calendar in the combo box.

Field Behavior

![[Date Field Behavior Interface of the Search Configuration.]][filtro_cons_data_hora_format] *Date Field Behavior Interface of the Search Configuration.*

- **Position between values** : This option sets the position that objects will be displayed.
- **Text between values** : Text that will appear between values.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Special Conditions

![[Datetime Field Special Conditions.]][filtro_cons_data_cond] *Datetime Field Special Conditions.*

- **All Period** : Searches for all periods of dates.
- **Today** : Searches in todays date.
- **Yesterday** : Searches in yesterdays date.
- **Last 7 days** : Searches the last 7 days. Ex: ((01/01/2017 01/07/2017).
- **This month** : Searches the dates from the first day of the current month.
- **Last month** : Searches the dates from the first day of lasts month.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).

- **Width** : To define a width for the field.
- **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

1. [Home](#)
2. [Applications](#)
3. Grid
4. Select

Select

General Settings

![[Select field Configuration Interface.]][[filtro_cons_select] *Select field Configuration Interface.*

- **Data Type** : Define the type of field for the application. When set to Select, you can select multiple option from a combo box (Select Field).
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Field Behavior

![[Select Field Behavior Interface of the Search Configuration.]][[filtro_cons_cpf_filtro] *Select Field Behavior Interface of the Search Configuration.*

- **Use autocomplete** : The field behaves as an autocomplete according to the values existing in the database.
- **OnChange Submit** : When enabled, it submits the search when there is modifications to the field.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Search Lookup

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

• Lookup Methods

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

Selecting the lookup type.

◦ Lookup Method - Automatic

Lookup used to list the values that will be displayed on the Checkbox field. These values will be recovered dynamically using the Select command.

Automatic Lookup Interface.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimitation.

- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter than (;).
- **Height** : Defines the height for the select object.
- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

- **Link** : Allows to create a link to another form allowing to manipulate the list displayed on the select field. After the manipulation, the select object it updated automatically.

◦ Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

Manual Lookup Interface.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).

- **Single Value** :

Setting up Single Value Lookup

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Default** : With this option enabled, the selected item will be inserted to the table when committed.
- **Height** : Defines the height for the select object.

■ Multiple Values (delimiter)

You can store various values for the select field. The values are separated by the informed delimiter. For example: the combination **Sport, Cinema and Tourism** selected in form will be stored like **E;C;T** in case the delimiter is ; (semi coma).

Setting up Multiple Values (delimiter)

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Height** : Defines the height for the select object.

■ Multiple Values (position)

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man, Single** and **Read**, in the database would be stored the following value **MSRD**.

Setting up Multiple Values (position)

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Negative** : Value to be saved in the table in case there is not any valued selected on the field.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.
- **Height** : Defines the height for the select object.

■ Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports, Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

Editing Lookup Configuration Multiple Values (Binary)

- **Label** : Text that will be displayed in the item list of the select field.
- **Height** : Defines the height for the select object.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.

- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

 Displaying the original lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Lookup Method - Actual value

This lookup is used to list all the values in the selected field.

This lookup will apply a “distinct” to your SQL query.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

1. [Home](#)
2. [Applications](#)
3. Grid
4. Double Select

Double Select

General Settings

![[Double Select field Configuration Interface.]]`[filtro_cons_duplo_select]` *Double Select field Configuration Interface.*

- **Data Type** : Define the type of field for the application. When set to Double Select, your allowed to have multiple options selected.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Field Behavior

![[Double Select Field Behavior Interface of the Search Configuration.]]`[filtro_cons_cpf_filtro]` *Double Select Field Behavior Interface of the Search Configuration.*

- **Use autocomplete** : The field behaves as an autocomplete according to the values existing in the database.
- **OnChange Submit** : When enabled, it submits the search when there is modifications to the field.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Search Lookup

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or F can be presented like Male or Female or in a dynamic way, getting these values from the database.

Lookup Settings Display for the field. *Automatic Lookup Interface..*

- **SQL Select Statement** : Defines the SQL command that will get the values displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the `key_field` will be stored in the table field.
- **Height** : Set the height(lines) of the field interface.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter than (;).
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Selects another specific connection existing in the project. The select command will be done on the second connection.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.

- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the Input Object.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
- **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the Input Object.
- **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

1. [Home](#)
2. [Applications](#)
3. Grid
4. Check box

Check box

General Settings

![[Check box field Configuration Interface.]][[filtro_cons_checkbox] *Check box field Configuration Interface.*

- **Data Type** : Define the type of field for the application. When set to Check box, your allowed to have multiple options selected.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Field Behavior

![[Check box Field Behavior Interface of the Search Configuration.]][[filtro_cons_cpf_filtro] *Check box Field Behavior Interface of the Search Configuration.*

- **Use autocomplete** : The field behaves as an autocomplete according to the values existing in the database.
- **OnChange Submit** : When enabled, it submits the search when there is modifications to the field.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Search Lookup

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or F can be presented like Male or Female or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

• Lookup Methods

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).
Selecting the lookup type.

◦ Lookup Method - Automatic

Lookup used to list the values that will be displayed on the CheckBox field. These values will be recovered dynamically using the Select command.

Automatic Lookup Interface.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.

- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Option check all** : Displays two options on the field to check and uncheck all.

- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Selects another specific connection existing in the project. The select command will be done on the second connection.

◦ Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

Manual Lookup Interface.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).

▪ Single Value :

Setting up Single Value Lookup

- **Label** : Text that will be displayed in the item list of the checkBox.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Default** : With this option enabled, the selected item will be inserted to the table when committed.
- **Columns** : Set the amount of columns, for the list of items.

▪ Multiple Values (delimiter)

You can store various values for the checkBox field. The values are separated by the informed delimiter. For example: the combination **Sport, Cinema and Tourism** selected in form will be stored like **E;C;T** in case the delimiter is ; (semi coma).

Setting up Multiple Values (delimiter)

- **Label** : Text that will be displayed in the item list of the checkBox.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will

view in the list **Man**.

- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Columns** : Set amount of columns, for the list of items.

▪ Multiple Values (position)

Stores a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in the database would be stored the following value **MSRD**.

Setting up Multiple Values (position)

- **Label** : Text that will be displayed in the item list of the checkBox.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Negative** : Value to be saved in the table in case there is not any valued selected on the field.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.
- **Columns** : Set the amount of columns, for the list of items.

▪ Multiple Values (binary)

Stores a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute Value	Lookup Description
1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value	Lookup Description
1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value	Lookup Description
1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

Setting up Multiple Values (Binary)

- **Label** : Text that will be displayed in the item list of the checkBox.
- **Columns** : Allows you to inform the amount of columns, for the list of items.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Saves all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Refreshes the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Option check all** : Displays two options on the field to check and uncheck all.

- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Grid
- 4. Radio

Radio

General Settings

![[Radio field Configuration Interface.]][[filtro_cons_radio]] *Radio field Configuration Interface.*

- **Data Type** : Define the type of field for the application. When set to Radio, your allowed to select one of the options listed.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Field Behavior

![[Radio Field Behavior Interface of the Search Configuration.]][[filtro_cons_cpf_filtro]] *Radio Field Behavior Interface of the Search Configuration.*

- **Use autocomplete** : The field behaves as an autocomplete according to the values existing in the database.
- **OnChange Submit** : When enabled, it submits the search when there is modifications to the field.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Search Lookup

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

• Lookup Methods

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

Selecting the lookup type.

◦ Lookup Method - Automatic

Lookup used to list the values that will be displayed on the Checkbox field. These values will be recovered dynamically using the Select command.

Automatic Lookup Interface.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.

- **Columns** : Allows you to inform the amount of columns, for the list of items.

- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

◦ Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

Manual Lookup Interface.

- **Label** : Text that will be displayed in the item list of the radio.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Default** : With this option enabled, the selected item will be inserted to the table when committed.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

Displaying the original lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

1. [Home](#)
2. [Applications](#)
3. Grid
4. Text Auto-Complete

Text Auto-Complete

General Settings

![Text Auto-Complete field Configuration Interface.][filtro_cons_texto_auto] *Text Auto-Complete field Configuration Interface.*

- **Data Type** : Define the type of field for the application. When set to Text auto complete, you can inform a value based on the select statement of the lookup settings and it will manipulate an internal Text for the data.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Field Behavior

![Text Auto-Complete Field Behavior Interface of the Search Configuration.][filtro_cons_texto_filtro] *Text Auto-Complete Field Behavior Interface of the Search Configuration.*

- **Use autocomplete** : The field behaves as an autocomplete according to the values existing in the database.
- **Case Settings** : Allows to convert the letters of the field when losing the focus. The options are:
 - **Upper Case** : All in Upper Case.
 - **Lower Case** : All in Lower Case.
 - **Capitalize first word** : Capitalizes the first letter of the word.
 - **Capitalize all words** : Capitalizes the first letter of all the words.
- **OnChange Submit** : When enabled, it submits the search when there is modifications to the field.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Search Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field.
- **Rows** : Allows to define the amount of records displayed in the field.
- **Width** : Defines the width the size of capture box (Capture Text).
- **Search options** : Allows to define the search settings of the field(Start equal to, Any part and End equals to).
- **Capture Text** : When not enabled, you will only view the field to inform the data. See an example below. *lookup Settings configuration Interface.*
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

 lookup Settings validation configuration Interface.
- **Show label with the description** : Displays a label with the description.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.

- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the Input Object.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
- **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the Input Object.
- **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

1. [Home](#)
2. [Applications](#)
3. Grid
4. Number Auto-Complete

Number Auto-Complete

General Settings

![[Number Auto-Complete Field Behavior Interface of the Search Configuration.]][[filtro_cons_numero_auto]] *Number Auto-Complete Field Behavior Interface of the Search Configuration.*

- **Data Type** : Define the type of field for the application. When set to Number auto complete, you can inform a value based on the select statement of the lookup settings and it will manipulate an internal number for the data.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Field Behavior

![[Number Auto-Complete Field Behavior Interface of the Search Configuration.]][[filtro_cons_decimal_filtro]] *Number Auto-Complete Field Behavior Interface of the Search Configuration.*

- **Use autocomplete** : The field behaves as an autocomplete according to the values existing in the database.
- **Position between values** : This option sets the position that objects will be displayed.
- **Text between values** : Text that will appear between values.
- **OnChange Submit** : When enabled, it submits the search when there is modifications to the field.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Values format

![[Number Auto-Complete Field Format of Values with Regional Settings.]][[filtro_cons_decimal_format]] *Number Auto-Complete Field Format of Values with Regional Settings.*

- **Regional Settings** : Allows to apply the Regional Settings to format the number of the field. When not enabled, the attributes Digit Grouping Separator, Decimal Separator, Negative sign and Negative number format.
- **Decimal Precision** : Defines the amount of decimal places for the field.
- **Digit grouping** : Defines if the field will display the digits separator.
- **Decimal Separator** : Defines the decimal separator.
- **Accept** : Determines if the field, will accept only negative, positive or both numbers.

Search Lookup

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field.
- **Rows** : Allows to define the amount of records displayed in the field.
- **Width** : Defines the width the size of capture box (Capture Text).
- **Search options** : Allows to define the search settings of the field(Start equal to, Any part and End equals to).
- **Capture Text** : When not enabled, you will only view the field to inform the data. See an example below. *lookup Settings configuration Interface.*
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup. *lookup Settings validation configuration Interface.*
- **Show label with the description** : Displays a label with the description.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.

- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
- **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the field.
- **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

Settings

With this interface, you can set the common attributes of the app.

ATTRIBUTE	VALUE
Application Code	grid_customers (9.00.0000)
Description	<input type="text"/>
Documents Path	C:/Program Files/NetMake/v9/www/
Image Directory	/scriptcase/file/img
Application images	<input type="button" value="Add"/> <input type="button" value="Remove"/>
Language	English (United States) ▾
Share Location Variable	<input checked="" type="checkbox"/>
Charset	<input type="text"/>
Share Theme Variable	<input checked="" type="checkbox"/>
Folder	root ▾
Edit by Project	<input checked="" type="checkbox"/>
Timeout	<input type="text" value="0"/>
HelpCase Link	Application ▾ Search ▾ Summary ▾

Application Settings Interface

- Attributes**
 - Application Code** : It is the name that defines an application. An app can be renamed at the [List of Application](#).
 - Description** : This field contains a brief description of the application objectives.
 - Documents Path** : The absolute path to store uploaded documents in the application.
 - Image Directory** : The filesystem directory to store the application images.
 - Application images** : Import images into the application to allow using them in the application.
 - Language** : Set the default language of the application. Display all the application hints and messages in the selected language.
 - Share Location Variable** : Define if the app shares the regional settings with other applications through a session variable.
 - Charset** : Define a specific charset to use in the application.
 - Share Theme Variable** : Define if the app shares the Theme settings with other applications through a session variable.
 - Folder** : Define the project folder that contains the app.
 - Edit by Project** : Define if other project developers can edit the application.
 - Timeout** : Set the session runtime timeout in seconds. If the value is Zero, it assumes the default timeout of the PHP.
 - HelpCase Link** : It allows to associate a [HelpCase](#) file with the application.

Notification Settings

ATTRIBUTE	VALUE
Use SweetAlert	<input checked="" type="checkbox"/>
SweetAlert position	Center ▾
Script Error	<input type="checkbox"/>
SQL Error	<input checked="" type="checkbox"/>
Debug Mode	<input type="checkbox"/>
Ajax Error Output	<input checked="" type="checkbox"/>

- Use SweetAlert**: Use the SweetAlert to display messages from the application. When this option is active, it will replace the browser's "confirm" and "alert".
- SweetAlert position using Toast** : The position to display error messages on the application.
- Script Error** : Allows displaying the line code where there is an error..
- SQL Error** : Allows displaying the SQL statement if it got an error.
- Debug Mode** : Runs the application in Debug mode, showing all SQL statements the application is executing.
- Ajax Error Output** : Enables the Ajax alert for debugging errors.

Navigation

This interface allows defining the navigating behavior of the application

ATTRIBUTE	VALUE
Exit URL	<input type="text"/> link
Close on Exit	<input type="checkbox"/>
Redirect URL	<input type="text"/> link
Redirect Variable	<input type="text"/>

Navigation Interface.

Exit URL

URL to where the user goes when he clicks on the "exit" button.

Close on Exit

Close the browser window when the user clicks on the "exit" button.

Redirect URL

Redirect to another URL in case there aren't any global variables available.

Redirect Variable

Creates a variable with the application name and sends it to the redirected application.

Global Variables

This screen shows all global variables used in the application

The global variable is an external parameter required for running the application. The application can include global variables in the WHERE clause, field definitions and names, event programming, and so on.

You can define global variables by using square brackets ([variable]). You can pass the parameters to the application through one of the methods: Session, Post, and Get

if you need to use [Database Identifiers](#) in or tables, we recommend using the double quotation marks(" ") instead of the brackets([]) because of the conflict with the [Global Variables syntax](#) of Scriptcase. Using the brackets as database identifiers may lead to problems in the Scriptcase applications.

- Settings
- Navigation
- Messages
- Global Variable

Global variables Interface.

Example:

```
Select CustomerID, CustomerName from Customers WHERE CustomerID = '[v_customerid]'
```

We do not recommend using session variables (global variables) with the same name as the table fields.

In this case, the variable **v_customerid** is displaying in the global variables configuration.

ATTRIBUTE	VALUE
global	Scope <input type="checkbox"/> SESSION <input checked="" type="checkbox"/> POST <input checked="" type="checkbox"/> GET Settings <input type="checkbox"/> Optional Type <input type="radio"/> Out <input checked="" type="radio"/> In

Global variables configuration Interface.

Attribute

It shows the variable's name in the application.

Value

Defines the behavior of the variables, divided into three blocks, they are:

Scope

Defines how the application receives the variable. A variable defined as the POST method in the app does not receive value if it comes with the GET method. Those are the methods available:

- **ession**: Defines that the variable must be created as a PHP session variable by another application.
- **ET**: Defines that the variable must be passed by the URL, that is, visible by the browser.
- **OST**: Defines that the variable must be passed through the POST method of an HTML form.

Settings

- **ptional**: Check it to don't validate the variable at runtime.

Type (In / Out)

Defines whether the variable comes from another application (In), or it is starting in the current app (Out).

Description

Indicates where the application is using the variable.



1. [Home](#)
2. Applications

General Overview

Allows to create link between applications that are in the same project, the link option incorporates various functionalities like above.

New Link

To create a new link you just need to access the option **Link between Applications** located in the menu at the left side.

When clicking in **New Link** the screen below will be display:

When clicking in **New Link** the screen to choose the type of link that will be created will be displayed.

[Edit Link:](#)

Allows to edit registers of the grid using a form.

[Capture Link:](#)

Allows to create a link specifically to the filter's fields, capturing data from other grid.

[Field Link:](#)

Allows to create a link represented through a link, from a grid's column to any application from the project.

[Button Link:](#)

Allows to create a link through grids buttons to any application from the project.

1. [Home](#)
2. [Applications](#)
3. Grid
4. Edit Link

Edit Link

Creating an Editing Link

This type of link allows the developer to create a link from a grid to a form with the objective of editing the register of a grid's row.

In the link options, we will choose the **Application Link**. When we choose this option, edit a register from a Grid will be possible.

Applications list

After we select this option, will be displayed the applications list to what you want to create the link.

This screen can be viewed from the following ways:

All:

In which is possible to see all the project's applications. **Example:**

By folder:

In which is possible to see the applications according to the folders in which each of them are.

By type:

In which is possible to see the applications grouped by its respective types.

Definition of Parameters

In this setting we are going to define the type of parameter that will be passed to the next application.

We have two options at **Type of Parameters**, they are:

Fixed value:

This option allows the user to define a fixed value that will be used to the call of the next application.

No value:

This option allows to create the link without the need to send any parameter to the next application.

It is also possible to see a refresh button at its right side.

This button should be used when a new parameter is added to the target application, so the new parameter can be loaded in the current application to make the link.

Link properties

In this screen we will set the application display mode that will be called in the link.

There are five display options, they are:

Open in the same window:

When we use this option the target application will be displayed in the same window of the application, and the target application will have a back button so we can return to the previous application.

Open in another window:

When we use this option the target application will be displayed in other browser window, and the target application will have a exit button so we can close this window.

Open in an iframe:

When we use this option the target application will be displayed in the same window of our application, however it will be displayed in a iframe in one of the four options available in the iframe settings.

Open in a parent:

When we use this option the target application will be displayed in the same window of our application, and the target application will have a back button so we can return to the previous application.

Modal:

When we use this option the target application will be displayed in the same window of our application, however it will be displayed in a small window above the current application.

Open in the same window.

When we use this option the target application will be displayed in the same window of the application, and the target application will have a back button so we can return to the

previous application.

When we select this option, we can set these following options:

Exit URL for the target application:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Display the button new in the grid:

When we select the option **Yes**, the button **New** is created in the grid to insert new registers.

Also, when we select the option **Yes**, it is displayed two new options **Label to the New button** and **Hint to the New button**:

Label to the New button:

In this option we will inform a name for the button that will be displayed in the Grid.

Hint to the New button:

In this option we will inform a message that will be displayed when the mouse cursor is over the **New** button.

Shortcut key to the New button:

Indicates the shortcut key to the button add new register.

Open in another window

When we use this option the target application will be displayed in another browser window.

When we select this option, we can set the following options:

Exit URL for the target application:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Close the form after an update:

Choosing the option **Yes**, the window will be closed after doing the update of a register.

Close the form after an insert:

Choosing the option **Yes**, the window will be closed after doing the insert of a register.

Open the application in a tab inside the menu:

Choosing the option **Yes**, the application will be open in a **menu tab**, when executed from a menu application, instead of opened in another browser window.

Display the button New in the grid:

When we select the option **Yes**, the button **New** is created in the grid to insert new registers.

Shortcut key to the New button:

Indicates the shortcut key to the button add New Register.

Label to the New button:

In this option we will inform a name for the button that will be displayed in the Grid.

Hint to the New button:

In this option we will inform a message that will be displayed when the mouse cursor is over the **New** button.

Open in an iframe.

When we use this option the target application will be displayed in the same window of our application, and the target application will have a back button so we can return to the previous application.

When we select this option, we can set the following options:

Shortcut key to the New button:

Indicates the shortcut key to the button add New Register.

Exit URL for the target application:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Iframe properties

In this settings, we can define some iframe details that will display the target application.

Display the target application's header:

When the option **Yes** is selected, the header of the target application is also displayed in the iframe.

Iframe position in relation to the main application:

In this option, we will define in which position in relation to the main application the iframe will be displayed, there are four options:

Below: The iframe will be displayed below the main application.

Above: The iframe will be displayed above the main application.

Right: The iframe will be displayed on the right of the main application.

Left: The iframe will be displayed on the left the main application.

Action after an insert:

In this option, we will define what will happen after a register insert, there are two options:

Reload the grid: The current page will be refreshed after the insert.

Move to the end of the grid: After the insert will be displayed the last page of the grid will the last register inserted.

Iframe height:

Allows to set the iframe height.

Iframe width:

Allows to set the iframe width.

Modal

When we use this option the target application will be displayed in the same window of the application, however she will be displayed in a small window above the current application.

When we select this option, we can set the following options:

Height:

Allows to set the Modal height.

Width:

Allows to set the Modal width.

Exit URL for the target application:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Close the form after an update:

Choosing the option **Yes**, the window will be closed after doing the update of a register.

Close the form after an insert:

Choosing the option **Yes**, the window will be closed after doing the insert of a register.

Display the button New in the grid:

When we select the option **Yes**, the button **New** is created in the grid to insert new registers.

Shortcut key to the New button:

Indicates the shortcut key to the button add New Register.

Label to the New button:

In this option we will inform a name for the button that will be displayed in the Grid.

Hint to the New button:

In this option we will inform a message that will be displayed when the mouse cursor is over the **New** button.

Form Properties

In those settings, we can define which buttons will be available in the target form application. Initially we have five options, that are:

Enable Insert button:

In this option we can define if the **New** button will be available in the target application.

Enable Update button:

In this option we can define if the **Save** button will be available in the target application.

Enable Delete button:

In this option we can define if the **Delete** button will be available in the target application.

Enable Navigation buttons:

In this option we can define if the buttons **first, previous, next and last** will be available in the target application.

Enable register editing button in the grid:

In this option we can define if the edit register button, which is the **pencil** in the grid, will be available.

1. [Home](#)
2. [Applications](#)
3. Grid
4. Field Link

Field Link

Creating a field link

Allows to create a link, represented through a link, from a grid's column to any project's existing application. All the links are displayed in a dropdown if there are more than one link to the same field.

In the types of link option, we will choose the **Field link**.

Applications list

After we select this option, will be displayed the applications list to what you want to create the link.

This screen can be viewed from the following ways:

All:

In which is possible to see all the project's applications. **Example:**

By folder:

In which is possible to see the applications according to the folders in which each of them are.

By type:

In which is possible to see the applications grouped by its respective types.

Definition of Parameters

In this setting we are going to define the type of parameter that will be passed to the next application.

We have two options at **Type of Parameters**, they are:

Fixed value:

This option allows the user to define a fixed value that will be used to the call of the next application.

No value:

This option allows to create the link without the need to send any parameter to the next application.

It is also possible to see a refresh button at its right side.

This button should be used when a new parameter is added to the target application, so the new parameter can be loaded in the current application to make the link.

Link properties

In this screen we will set the application display mode that will be called in the link.

There are five display options, they are:

Open in the same window:

When we use this option the target application will be displayed in the same window of the application, and the target application will have a back button so we can return to the previous application.

Open in another window:

When we use this option the target application will be displayed in other browser window, and the target application will have a exit button so we can close this window.

Modal:

When we use this option the target application will be displayed in the same window of our application, however it will be displayed in a small window above the current application.

Open in the same window

When we use this option the target application will be displayed in the same window of the application, and the target application will have a back button so we can return to the previous application.

When we select this option, we can set the following options:

Exit URL for the target application:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Hint for the link:

In this option we will inform a message that will be displayed when the mouse cursor is over the button with the link.

Open in another window

When we use this option the target application will be displayed in another browser window.

When we select this option, we can set the following options:

Exit URL for the target application:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Close the form after an update:

Choosing the option **Yes**, the window will be closed after doing the update of a register.

Close the form after an insert:

Choosing the option **Yes**, the window will be closed after doing the insert of a register.

Hint for the link:

In this option we will inform a message that will be displayed when the mouse cursor is over the button with the link.

Open the application in a tab inside the menu:

Choosing the option **Yes**, the application will be open in a **menu tab**, when executed from a menu application, instead of opened in another browser window.

Open in an iframe

When we use this option the target application will be displayed in the same window of our application, and the target application will have a back button so we can return to the previous application.

When we select this option, we can set the following options:

Exit URL for the target application:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Hint for the link:

In this option we will inform a message that will be displayed when the mouse cursor is over the button with the link.

Iframe properties

In this settings, we can define some iframe details that will display the target application.

Iframe position in relation to the main application:

In this option, we will define in which position in relation to the main application the iframe will be displayed, there are four options:

Below: The iframe will be displayed below the main application.

Above: The iframe will be displayed above the main application.

Right: The iframe will be displayed on the right of the main application.

Left: The iframe will be displayed on the left the main application.

Iframe height:

Allows to set the iframe height.

Iframe width:

Allows to set the iframe width.

Modal

When we use this option the target application will be displayed in the same window of the application, however she will be displayed in a small window above the current application.

When we select this option, we can set the following options:

Height:

Allows to set the Modal height.

Width:

Allows to set the Modal width.

Exit URL for the target application:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Close the form after an update:

Choosing the option **Yes**, the window will be closed after doing the update of a register.

Close the form after an insert:

Choosing the option **Yes**, the window will be closed after doing the insert of a register.

Hint for the link:

In this option we will inform a message that will be displayed when the mouse cursor is over the button with the link.

Grid's Properties

In those settings we can define how our target grid will be displayed. We initially have six options, that are:

Initial module:

We can define how our application will be executed in those two options:

Grid: The application will be executed as a grid.

Search: The application will be executed as a search.

Quantity of rows:

In this option we can define the quantity of rows that will be displayed in the target application.

Quantity of Columns:

In this option we can define the quantity of columns that will be displayed in the target application.

Pagination:

In this option we can define if the target application's pagination will be total or partial, in case partial is the selected option the quantity of rows will be the informed previously.

Enable header:

In this option we can define if the target application's header will be displayed.

Enable Navigation buttons:

In this option we can define if the buttons **first, previous, next and last** will be available in the target application.

1. [Home](#)
2. [Applications](#)
3. Grid
4. Capture Link

Capture Link

Applications list

After we select this option, will be displayed the applications list to what you want to create the link.

This screen can be viewed from the following ways:

All:

In which is possible to see all the project's applications. **Example:**

By folder:

In which is possible to see the applications according to the folders in which each of them are.

By type:

In which is possible to see the applications grouped by its respective types.

Definition of Parameters

In this setting we are going to define the type of parameter that will be passed to the next application.

We have two options at **Type of Parameters**, they are:

Fixed value:

This option allows the user to define a fixed value that will be used to the call of the next application.

No value:

This option allows to create the link without the need to send any parameter to the next application.

It is also possible to see a refresh button at its right side.

This button should be used when a new parameter is added to the target application, so the new parameter can be loaded in the current application to make the link.

Application with no parameters

However, when the target application does not have any defined parameters, the following screen is displayed:

Clicking in the button, you will be taken to the target application to create a parameter, so you can use the update button in the **parameters definitions** to refresh them.

Link properties

In this screen we will set the application display mode that will be called in the link.

In this type of link there is only one display option:

Modal:

When we use this option the target application will be displayed in the same window of our application, however it will be displayed in a small window above the current application.

Modal

When we use this option the target application will be displayed in the same window of the application, however she will be displayed in a small window above the current application.

When we select this option, we can set the following options:

Modal:

In this option we are going to define if the target application will be opened in a Modal.

Yes: This option will make the target application be opened in a modal. **No:** This option will make the target application be opened in a new window.

If **Yes** is selected in the previous option, the Modal **Height** and **Width** will be available.

Height:

Allows to set the Modal height.

Width:

Allows to set the Modal width.

If **No** is selected in the previous option, only those options will be available.

Allows to modify manually in the update:

When we select the option **Yes**, the button **New** is created in the grid to insert new registers.

Allows to modify manually in the insert:

Indicates which shortcut key to the button add new register.

Allows to modify automatically in the update:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Allows to modify automatically in the insert:

Choosing the option **Yes**, the window will be closed after doing the insert of a register.



1. [Home](#)
2. [Applications](#)
3. Grid
4. Button Link

Button Link




1. [Home](#)
2. [Applications](#)
3. Grid
4. Editando Ligações

Editando Ligações

Editando Ligações

In the Link Folder of the Application Menu (Image Below) are displayed the links existing in the application and also the item New Link. When clicking on the existing link it is displayed the screen below that allows to manage the links.

 *Editing Links.*

Actions

Properties Change the link's behavior, position, and how the link opens.

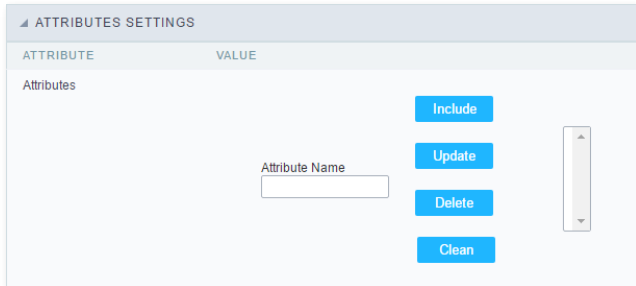
Link Change the application that's being called in the link their parameters.

Delete Remove the existing link.

In this version of ScriptCase is incorporated with the concept of programming with the use of attributes, methods, resources and libraries. In the previous versions it was already possible to create business rules in applications using this concept, but the big difference now is that this can be done in a more organized and dynamic way, facilitating both the development as the understanding of the rule by another developer.

Attributes

The attributes are variables that has the global scope declared in the application. An attribute that can be referenced in all the methods and the application's events.



Attributes configuration Interface.

- **attributes**
 - **attribute Name** : This field allows to inform the attributes name, after typing the name just click on **include**. In case you want to **update** click on the attributes name (located on the buttons' right side), do the necessary modifications and click on update. In case it is necessary to **delete** an attribute, it is necessary to do same process as the update, but when selecting the attribute, it is necessary to click on the delete button. The **clear** button, as the name suggests, it clears the list of all the attributes listed on the right.

Internal Libraries

To use the internal libraries just select the correspondent (Project Libraries for example). This process allows the methods existing in the library, to be visible in all the application's Events and Methods. See how to manage the libraries by [clicking here](#).

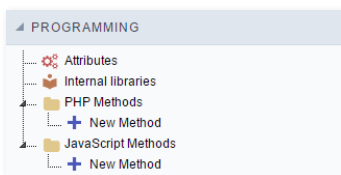


Internal Libraries management Interface

PHP Methods

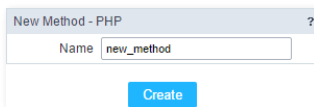
Methods are function or procedures declared by the developer, that helps when applying the routines. Using methods in the applications allows to reuse your code throughout the application, optimizing the development experience.

Creating A New Method



- Give the method a name and click Create. As in the image below.

PHP methods are incorporated into the same class, so it is not possible to use two functions with the same name even when used in different methods.



- Methods can receive parameters.

```
function new_method
1 echo "Hello World!!!";
```

- dd the amount of variables:

Definition of the parameters of the method:

new_method

- No defined parameter.

Add Parameter(s) Cancel

Save

- defining the variables:

Insertion of Parameters

Name	Type	Value Standard
<input type="text"/>	For Value	<input type="text"/>
	For Value	
	For References	

Save Back Cancel

- **ame** : Type in the variable's name.
- **ype** : Selecting the type of variables: For Value or For Reference.
- **alue Standard** : The parameter's value used to initialize when calling the method.

References allows to create a second name for a variable that you can use to read and modify the variable's original information.

- diting a parameter:

Definition of the parameters of the method:

new_method

Parameters

Stest = test

⏪ ⏩

☑ ☐ ✎ ✕

Add Parameter(s) Cancel

Save

- ☑ : Checking all the parameters in the parameter list.
- ☐ : Uncheck all the parameters in the parameter list.
- ✎ : Edit the selected parameter of the list.
- ✕ : Deletes the selected variable of the list.

It allows you to send the generated export file by email. For that, you may be using SMTP or integration with **Mandrill** and **SES**.

For more information about Mandrill [click here](#)

For more information about Amazon SES [click here](#)



Sending Options

We must configure how to send the email or select an API already set in Tools> API. [Click Here](#) and see how to configure.

SMTP

SMTP is the standard protocol for sending emails over the Internet, and each provider has its SMTP.

For more information about SMTP settings [click here](#)

Sending settings		
ATTRIBUTE	VALUE	DESCRIPTION
API	- Custom -  	API for sending e-mails.
Gateway	smtp	
SMTP server	smtp.example.com	SMTP server to send email.
SMTP port	465	SMTP server port (example: 25, 465, 587).
SMTP user	root	User to connect to the SMTP
SMTP password	*****	SMTP server connection password.
SMTP protocol		Encryption protocol used by the SMTP server.
From email	default@example.com	Default shipping email.
From name	default	Default shipping name.

- **PI:** Allows you to select an API that is already configured in **Tools> API** or set a new one here by selecting **- custom -**. In this case you will see some options according to the selected Gateway.

SMTP Server

SMTP server address for the used provider.

Port SMTP

Define the Port of the mail server. Use the port 465 for security with SSL, 587 for security with TLS, or port 25 for port without security. By default, Scriptcase uses port 25.

User SMTP

SMTP server user.

Enter SMTP

SMTP server user password.



Protocol SMTP

Select the security protocol. If no value is declared, Scriptcase uses the No Security protocol as default.

- **-mail:** Sets the origin email, which will send the emails.
- **ame:** Name that will be displayed in the sent email.

Mandrill

Mandrill is a transactional email API for MailChimp users, ideal for sending data-driven emails.

Sending settings		
ATTRIBUTE	VALUE	DESCRIPTION
API	- Custom -  	API for sending e-mails.
Gateway	mandrill	
API key	Your API	API connection key.
From email	default@example.com	Default shipping email.
From name	default	Default shipping name.

- **PI:** Allows you to select an API that is already configured in **Tools> API** or set a new one here by selecting **- custom -**. In this case you will see some options according to the selected Gateway.
- **PI KEY:** Enter the key you obtained when setting up your API.
- **-mail:** Sets the origin email, which will send the emails.
- **ame:** Name that will be displayed in the sent email.

Amazon SES

Amazon Simple Email Service ([Amazon SES](#)) is an email sending service designed to assist in sending marketing emails, notifications, and transactional messages.

Sending settings		
ATTRIBUTE	VALUE	DESCRIPTION
API	- Custom - <input type="button" value="🔄"/> <input type="button" value="✎"/>	API for sending e-mails.
Gateway	Amazon SES	
API key	Your Key API	API connection key.
API Secret	Your Secret API	API secret, obtained in association with API KEY.
Region	Region	API Connection Region.
From email	default@example.com	Default shipping email.
From name	default	Default shipping name.

- **PI**: Allows you to select an API that is already configured in **Tools> API** or set a new one here by selecting **- custom -**. In this case you will see some options according to the selected Gateway.
- **PI KEY**: Enter the key you obtained when setting up your API.

API Secret

Enter the secret access key of your Amazon SES account.

Region

Amazon SES has endpoints in several regions, to reduce network latency, inform the region of the endpoint closest to your application. [See the regions.](#)

- **-mail**: Sets the origin email, which will send the emails.
- **ame**: Name that will be displayed in the sent email.

Export Settings

In these settings, we define the settings of the email sending interface when exporting, and we can define a default email subject and body.

Export settings		
ATTRIBUTE	VALUE	DESCRIPTION
To	<input type="text" value="[mail]"/>	Enter the default value for the field 'To'.
It has copy	Yes	Sets whether to display the field 'copy' within exporting options.
Copy (cc)	<input type="text" value="[mailcopy]"/>	Enter a default value for the field 'Copy'.
It has blind carbon copy (bcc)	Yes	Sets whether to display the blind carbon copy field in export option.
Blind carbon copy	<input type="text" value="[mailbcc]"/>	Enter the default value for the field 'blind carbon copy'.
Title	<input type="text" value="{lang_export_email_subject} sales"/>	Default email subject text. You can change the lang variable or add fixed text.
Search of sales		
Body	<input type="text" value="{lang_export_email_body}"/>	Default text for the body of the email. You can change the lang variable or add a fixed text.
Hello, Please find attached with this email the report in %. Thank you and Have a Nice Day		

To

Enter the default destination email.

Use Copy (CC)

Defines whether or not to display the 'copy' field within the export options.

- **o**: There will be no possibility to send email with copies.
- **es**: The field will be displayed within the export options and we can define a default email for copying. The end user can view this email.
- **idden**: The field will not be displayed within export options and we can set a default email for copying.

Copy (CC)

Enter the default email for the copy field. This option can also be empty, or you can use a global variable containing an email list.

Using Hidden Copy (BCC)

Sets whether or not to display the 'hidden copy' field within the export options.

- **o**: There will be no possibility to send email with copies.
- **es**: The field will be displayed within the export options and we can define a default email for copying. The end user can view this email.
- **idden**: The field will not be displayed within export options and we can set a default email for copying.

Copy (BCC)

Enter the default email for the hidden copy field. This option can also be empty, or you can use a global variable containing an email list.

Subject

Default email subject text. You can change the lang variable or add static text.

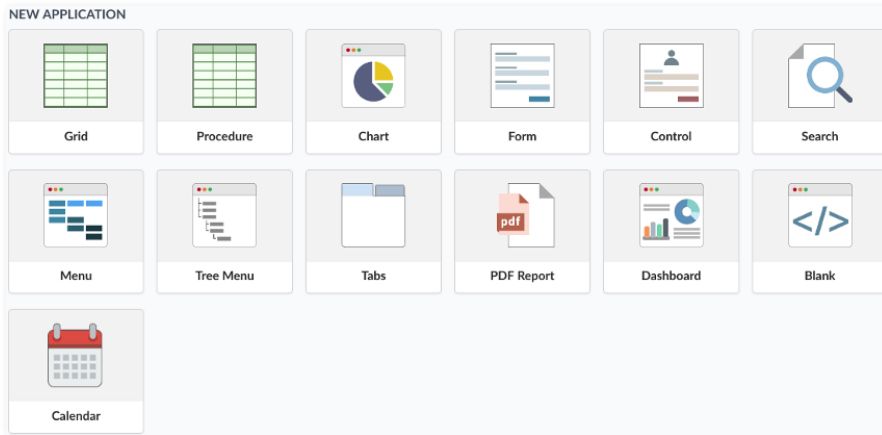
Color

Default body text for the email. You can change the lang variable or add static text.

New Application

The creation process of one application is simple enough and intuitive.

You just need to select which application you want to generate in the applications available in Scriptcase, That it will be generated automatically.



Application Data

When the application is selected, the following screen will be displayed below.

In this table you need to select the connection that will be used in the application. The tables will be loaded and displayed in the field **Table** when the connection is selected.

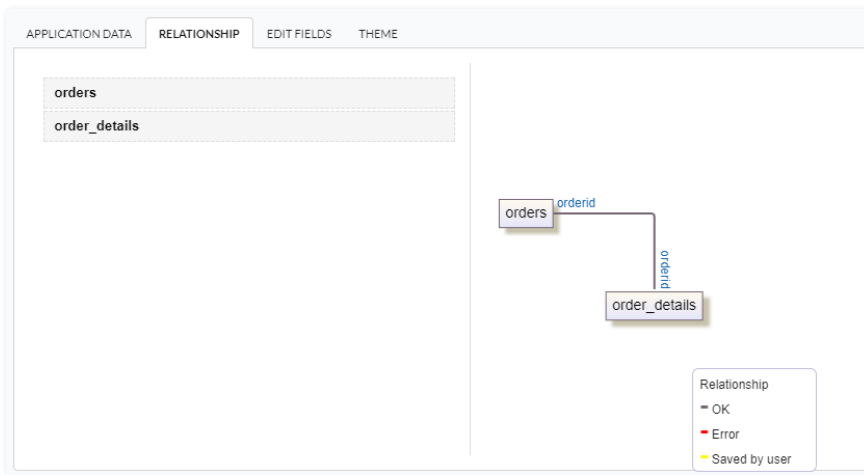
It is possible to select more than one table in the application creation.

After selecting the table and fields that should be part of the application, the name of the application will be formed by the type of application + name of the table. **Ex: grid_orders** However it is possible to change the application's name to any other desired name.

- **connection** - It defines the used connection to create the application. All the project's application will be displayed.
- **ables** - It defines the used tables in the application. (Form and Calendar can only use one table).
- **ields** - It defines the fields that will be part of the applications.
- **QL Select Statement** - Display the select statement created after selecting the tables and fields. This field allows also to insert a previously created SQL, if the used tables are available in the database selected in the connection.
- **ame** - Application's name that is being created, It can not contain special characters.
- **ocalization** - Application's Language. The project's default language will be automatically selected.

Relationship

When select two or more tables, the tab "Relationship" will be displayed. In this tab we can see the relationship created between the tables, where we can edit the related fields.



When we click in a link, in the screen above, it will displayed the related field's edition form, as you can see in the image below.

Edit Fields

In this tab will be displayed the fields that are going to be created in the application, also its labels and datatypes. We can make define some settings here like changing its labels and datatypes.

Fields	Label	Datatype	Grid	Search
orderid	Orderid	Integer	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
customerid	Customerid	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
employeeid	Employeeid	Integer	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
orderdate	Orderdate	Date	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
requireddate	Requireddate	Date	<input checked="" type="checkbox"/>	<input type="checkbox"/>
shippeddate	Shippeddate	Date	<input checked="" type="checkbox"/>	<input type="checkbox"/>

- **ields** - Name of fields stored in the database.
- **abel** - Name of fields displayed in the generated application.
- **atatype** - Field's Datatype.
- **rid** - It defines the available fields in the grid.
- **earch** - It defines the available fields in the search.

Theme

In this screen you select the theme that the application will use. The default project's theme, that can be checked and changed in **Project > Properties**, will be selected automatically.

APPLICATION DATA RELATIONSHIP EDIT FIELDS **THEME**

Sc9_Rhino ▾

Header

◀ ◀ ▶ ▶ Add Save

Block 1.1

Title 1 Object text

Block 2.1

Title 1	Title 2	Title 3
11111111	22222222	33333333
11111111	22222222	33333333

Footer

- **heme** - List of available themes in the project.

*After all the settings are done, click in the button **Create**. In the next screen the settings of the created application will be displayed.*

This application is designed for creating dynamic charts based on SQL or Procedures.

Inside the Scriptcase Charts can be created also within the Grid application, however the application Chart has more settings options to the end user within the application generated. The idea is to apply the concept of Business Intelligence so the final system users can apply their own settings within the generated charts.

The settings within the development area uses Drag and Drop for the charts creation. You can apply dimensions and metrics dynamically using different date periods and totals.

Dimensions

Dimensions are used to group the data into categories. You can choose more than one field as a dimension, each new field as a subcategory of the previous field.

To select a field, click on its name in the "Chart Fields" area and drag it to the "Dimensions" column.

Once selected the fields, you can sort them using the drag and drop to arrange the dimensions' final display. You can choose different periods to date fields and use the same date field more than once, with different periods. All the dimensions listed in "Dimensions" can be used by the system final user within the Chart application. You can also define whether dimensions will be displayed or not when you start the application by selecting the checkbox.

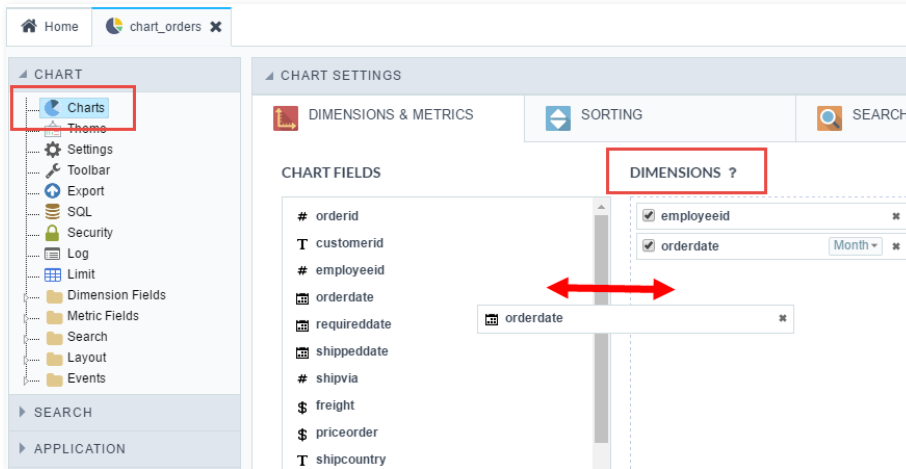
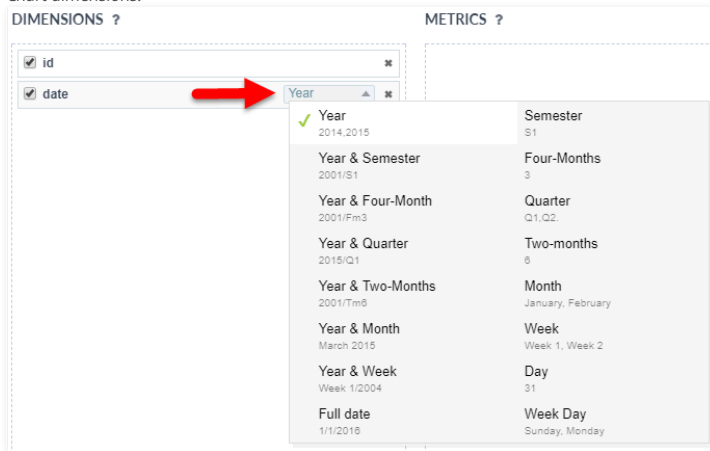


Chart dimensions.



Field settings

Metrics

The metric is used to quantitatively view the data grouped by dimensions. Choose the fields for the chart's metrics.

To select a field, click on its name in the "Chart Fields" area and drag it to the "Metrics" column. Once selected the fields, you can sort them using the drag and drop to arrange the metrics' final display, you can also chose the total function according to the field data type.

All the dimensions listed in "Metrics" can be used by the system final user within the Chart application. You can also define whether metrics will be displayed or not when you start the application by selecting the checkbox.

The checked field value is used to set the default field just when you are not using a combination or stacked chart types, that use more than one field. If you choose more then one field Scriptcase automatically sets a value multiple chart type (Combination). You can change the initial chart inside the option "Charts' type" within the "Settings". When you check a single value (bar, line, Spline, step, Area, pie, pyramid, funnel, Radar and Gauge) you have to click and choose only one field to appear as initial value during the first generation.

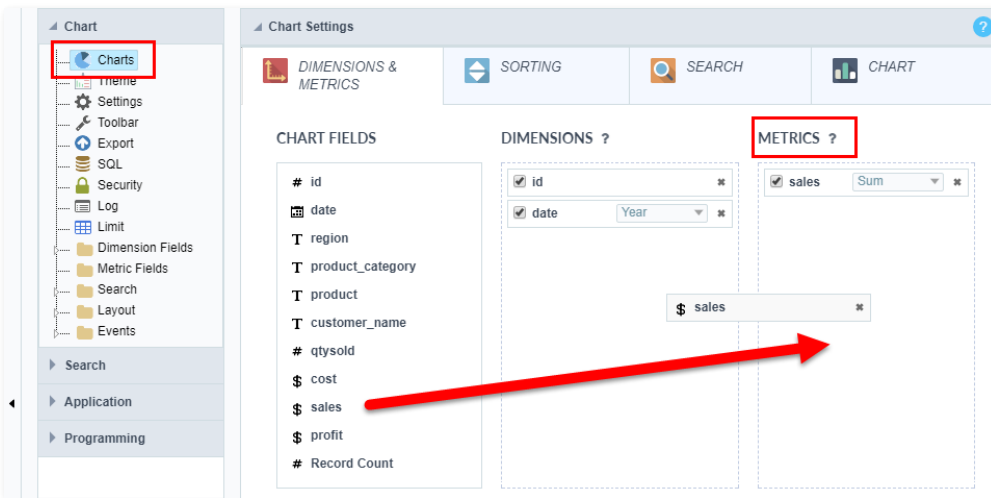


Chart Metrics.

Sorting

You can select which chart value will be ordered to start the application, and if the sort will be “ascending” or “descending”.

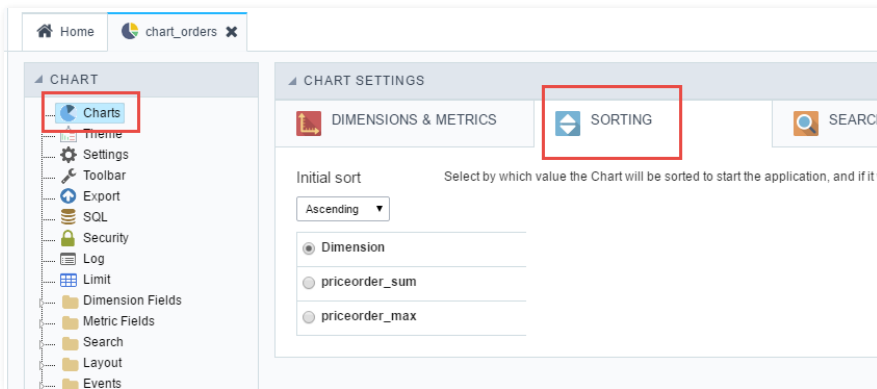


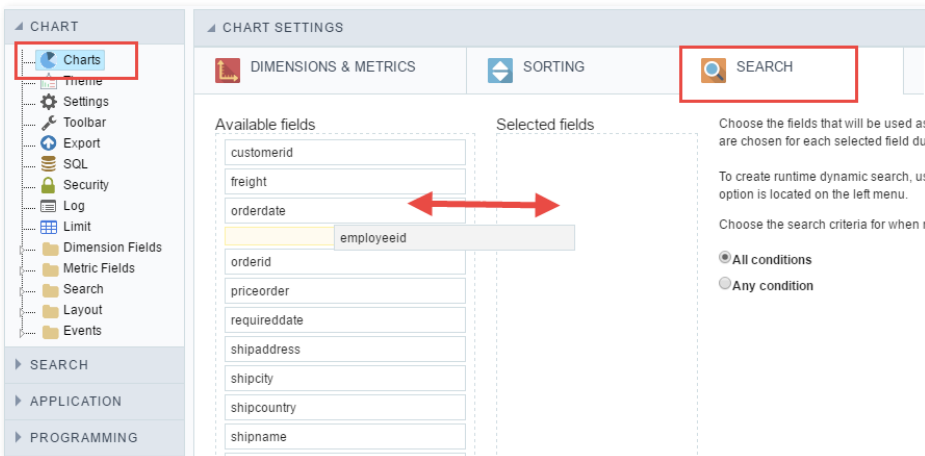
Chart sorting.

- **initial sorting** : Using this option you can set the default sorting (ascending or descending). The system final user will also be able to customize it within the generated application.
- **imension** : Application starts with dimension fields sorting (ascending or descending according to the “Initial sorting”).
- **etric** : Application starts with metric fields sorting (ascending or descending according to the “Initial sorting”).

Search

In the “Search” option you can choose the fields that will be used as static search on the chart, the application data will be initially filtered according to the settings you apply. The search values are chosen for each selected field during development, the final user will not be able to change that.

You are also able to apply other search options such as the Dynamic Search using the option “Search > Dynamic Search” (located on the left menu) and the “refined” search using the option “Chart > Search”. These two types are dynamic, it means that the end user will be able to manipulate them within the generated application.



Static search.

Chart options

In the tab "Chart" the developer sets other default options for the chart that will be generated such as chart type, color, theme and others. Find more details below.

Chart's type	2D Column
Export Chart	<input type="checkbox"/>
Font size	<input type="text"/>
Abbreviated value	<input type="checkbox"/>
Subtitle	Defined by the theme
Display Values	<input checked="" type="checkbox"/>
Exception for display	<input checked="" type="checkbox"/> Pie <input checked="" type="checkbox"/> Pyramid <input checked="" type="checkbox"/> Funnel
Chart Width	<input type="text" value="800"/>
Chart Height	<input type="text" value="600"/>
Values sorting	<input type="text"/>
Multi-series or Drill down	Drill down
Grand Total	<input type="checkbox"/>
Trendline New	<input type="text"/>
Chart trendline type New	Fixed
Initial value of the line New	<input type="text" value="0"/>
Final value of the line New	<input type="text" value="100000"/>
Bar - width (between 20 and 70)	<input type="text"/>
Bar - Value orientation	Vertical
Bar - Value position	Inside
Bar - Stacking	off
Pie - Values format	Value
Range of data	<input type="checkbox"/> Use interval
Pyramid - Values format	Value

Chart options.

- **tributes**

- o **raph Type**: Allows you to set the type of chart to be displayed by "Default" if the Configurable Chart option is active, or the type of chart to be displayed if the option is disabled.
- o **ize Size**: Font size of the values, if you leave the field blank, size 12 will be assumed.
- o **hort value**: Value displayed in the chart should be abbreviated.
- o **iew Values**: Starts already displaying the values of the generated graph.
- o **xception for Display**: Chart types that will not display chart values, even when marked to display. In these types we do not recommend that the values are displayed, for aesthetic reasons, in the meanwhile, it is up to the developer.
- o **raph Width**: Default Width in pixels of generated graphs.
- o **_ Graph Height_**: Default height in pixels of the generated graphs.
- o **rdination of values**: Sorts the values of the graphs.
- o **ulti-series or Drill down**: Defines whether the dimensions of the graph will be generated by grouping by multi-series or by drill down navigation.
- o **otal General**: If you want to group all the values in one, check the Total General option. This option is typically used in Gauge type graphics.
- o **_ Trend line_**: Defines a line or area for the trend line of the chart.
- o **rend line type**: Trend line type.
- o **_ Initial line value_**: Defines the initial value of the trend line.
- o **alue line end**: Sets the end value of the trend line.
- o **_ Bar - Width (between 20 and 70) _**: Width of the chart bar.
- o **ar - Value Orientation**: Orientation of the value displayed in the graph (Horizontal or Vertical).
- o **ar - Positioning of the value**: Positioning of the value in the graph (Above or Inside).
- o **ar - Stacking**: Stacks the bars in a single (By series).
- o **ie - Value format**: Format of the data in pie charts.
- o **ata Range**: Use data range with different colors.
- o **ramid - Values format**: Format that will display the value in the chart.

Chart settings

In the chart Setup screen the user can change some application viewing options, which charts will be displayed, the labels, margins and alignments. Below detailing each of the available attributes

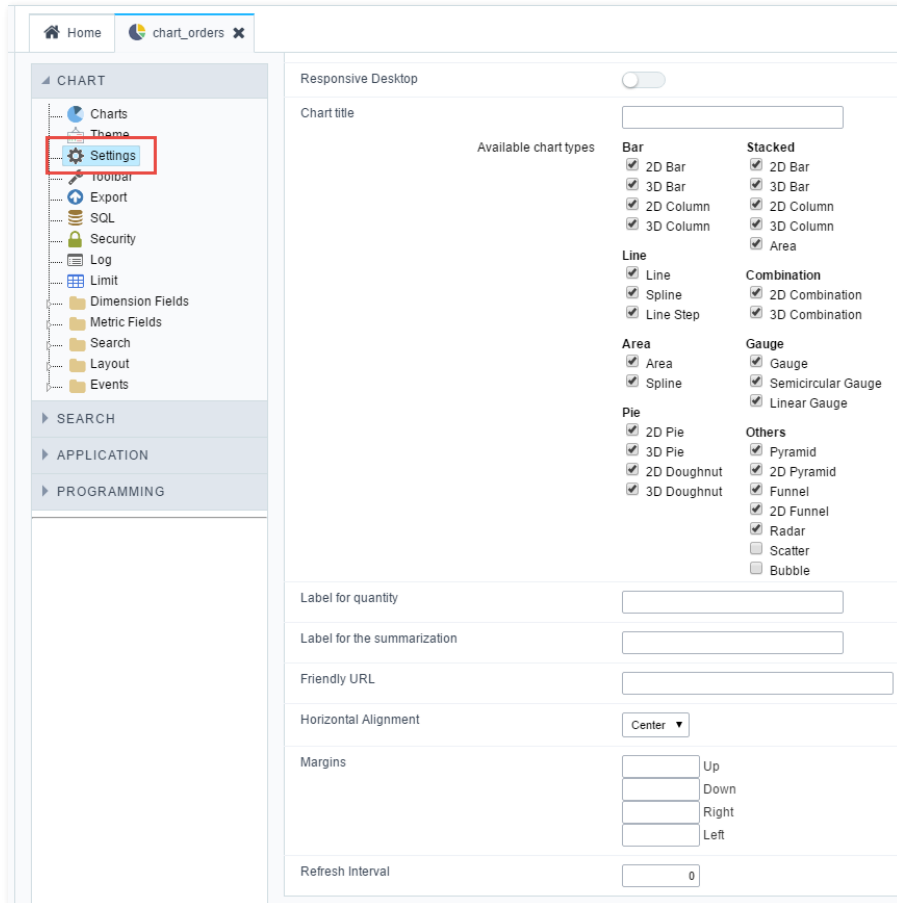


Chart settings interface

- **Attributes**
 - **Responsive Desktop** : By checking this option the Chart will automatically adapt itself to the user's screen according to the size of the browser.
 - **Chart title** : This option allows you to set a title for the chart.
 - **Available chart types** : List of available chart types for user's choice within the final application via the button "Chart Types". The graphics checked will be available, if the developer does not want to display a chart type must do not check them in this interface.
 - **Label for quantity** : Label the y-axis for generic values of the record count.
 - **Label for summarization** : Label the y-axis for generic values of the summary functions - sum, average, weighted average, maximum and minimum.
 - **Friendly URL** : This field allows you to change the URL that will be called by the application. Allowed characters are the same available on URLs: a-z, A-Z, 0-9, -_. This option can also be changed on the home screen, on the "Friendly URL" column at the applications list.
 - **Horizontal Alignment** : This option allows you to set the horizontal alignment of the application on page-centered, left or right
 - **Margins** : Sets the application margins in pixels (up, down, right and left).
 - **Refresh Interval** : Allows you to set a reload interval for the page, in seconds. When is set as zero, there will be no page reload.

The application toolbar has two segments: Top and Bottom, in a way that is possible to define to display buttons into both areas. Those areas work independently, allowing them to display the same button, for example.

It's also possible to select the buttons and their position if the application is running on a mobile device.

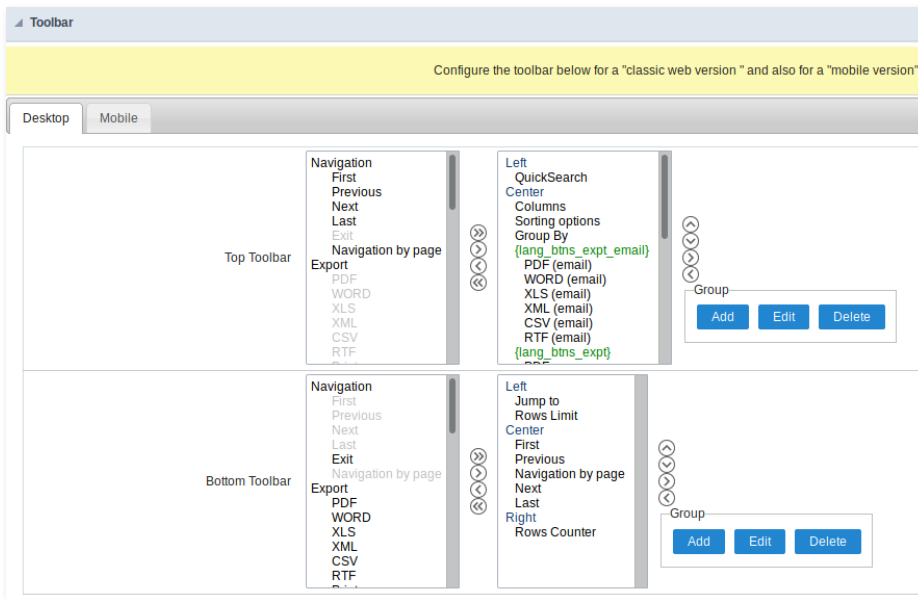
Toolbar

Desktop

Here we must inform the toolbar settings for the "Classic Web Version" mode and which buttons are available in the application when accessed from a **Desktop** environment.

Mobile

Here we must inform the toolbar settings for the "Mobile Version" mode. That is which buttons are available in the application when accessed from a **Mobile** dispositive.



Navigation:

Buttons relative to the navigation of the application.

Next	Move to the next page that can be a single record or a list of records.
Previous	Returns to displays the previous page records or a single record.
First	Move to the First page or record
Last	Move to the Last page or record
Exit	Close the application
Navigation by page	Displays a "page-number" navigation bar. Example: 1 2 3 4 5

Export:

Groups the options relative to the generated exports. Scriptcase generates the following export formats:

PDF	Generates a complete Report with all the data of the application in a PDF format.
WORD	Generates a complete Report with all the data of the application in a WORD format.
XLS	Generates a complete Report with all the data of the application in an EXCEL format.
XML	Generates a complete Report with all the data of the application in an XML format.
CSV	Generates a complete Report with all the data of the application in a CSV format.
RTF	Generates a complete Report with all the data of the application in an RTF format.
Print	Creates an HTML with the records ready for printing.

Export by Email:

Groups the options relative to the emails exports. Scriptcase generates the following export formats:


PDF (Email)	Sends by Email a complete report with all the data of the application in a PDF format.
WORD (Email)	Sends by Email a complete report with all the data of the application in a WORD format.
XLS (Email)	Sends by Email a complete report with all the data of the application in an EXCEL format.
XML (Email)	Sends by Email a complete report with all the data of the application in an XML format.
CSV (Email)	Sends by Email a complete report with all the data of the application in a CSV format.
RTF (Email)	Sends by Email a complete report with all the data of the application in an RTF format.

Others:

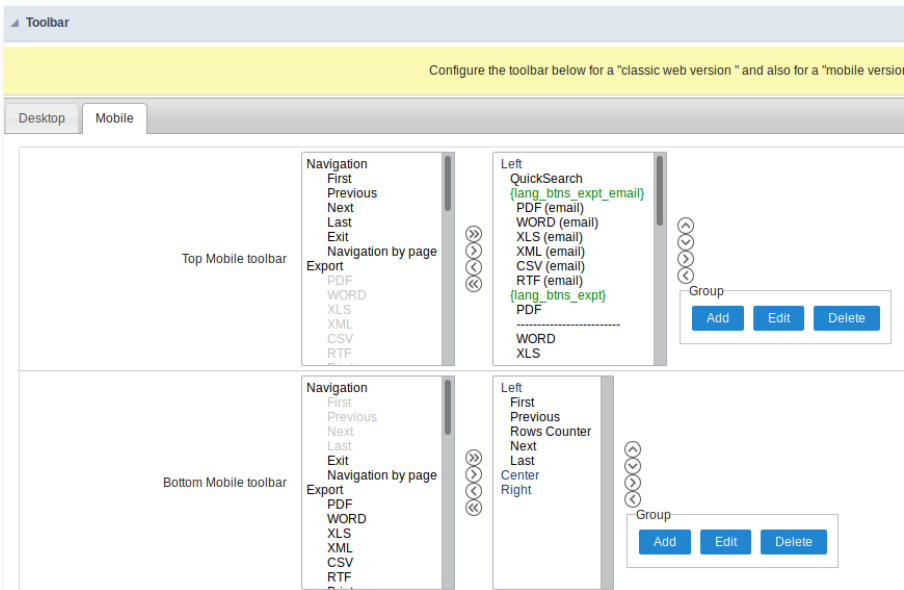
Other options available in the Grid application.

Jump to	Move to the informed page.
Rows Limit	It is a Combobox that defines the number of rows per page.
Search	Goes to the Search Form to filter the records.
Dynamic Search	It displays the fields of the search to filter the records.
Columns	Allows to include or remove columns of the Grid on the fly.
Sorting Options	Allows selecting the order of the records based on the field.
Group By	Allows to select or change a Group By rule on the Grid.
Save Grid	Allows saving the current state of the application. For example, in the advanced search, you can save the search data for further use.
Quick Search	Allows to perform a quick search in the records of the application.
Gantt	It displays a Gantt chart, if it was previously set.
Summary	It displays a summary with the synthetical data of the records.
Form Buttons	It displays the Form Buttons when there's an Application Link from the Grid to a Form.
Languages	It displays a Combobox with the names available, defined in the project properties.
Themes	It displays a Combobox with the themes available, defined in the project properties.
Rows Counter	It displays the number of records retrieved by the application.
HelpCase	Displays a button to open the help page.

Separator:

	Displays a line separating the buttons, when used the Group Buttons.
--	--

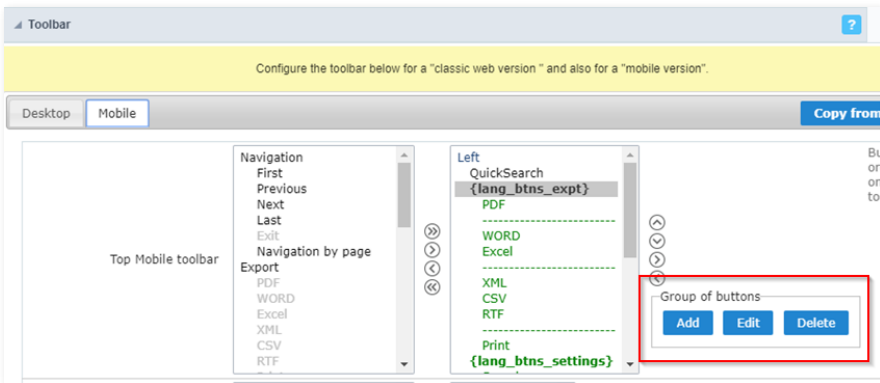
Toolbar Mobile



Contains the same options as the **Desktop** version, adding only the item "Copy from Desktop", on which, when clicked, it copies the items from the **Desktop** toolbar to the **Mobile** toolbar.

Buttons Group

The **Group** option allows you to group a set of buttons of the application toolbar to display them as a dropdown, for example.



Add

Add a new group of buttons.

Edit

Edit an existent group of buttons.

Delete

Delete the selected group of buttons.

When you press the **Add** or **Edit** option, you can see the settings to configure the grouper:

Edit

DISPLAY AS
 Dropdown

DROPDOWN LIST THEME NAME
 Application Theme group_4

LABEL HINT \ TITLE
 Group

IMAGE BUTTON TYPE
 Button

DISPLAY DISPLAY POSITION
 Only Text Text on the right

Cancel Add

Display As

Allows displaying the group button as **Dropdown** or **Side by Side**.

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print Exit

Customer ID	Company Name	Phone	Credit Limit
dos		30074321	\$3,367.41
arilla		55554729	\$7,371.95
		55553932	\$6,757.53

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print Exit

Dropdown List Theme

Allows defining the Dropdown theme selecting between **Application theme** and **Button theme**.

Name

Allows defining a name for the button group.

Label

It is the displayed name for the button group in the application.

Hint\Title

Displays a hint to the end-user when the mouse is on the group of buttons.

Button Type

Allows displaying the button group as a Button, Image, or Link.

Image

Allows selecting an image for the button.

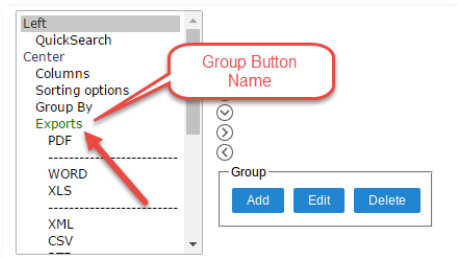
Display

Defines if the button displays only Text, only image, or both.

Display Position

Defines the position of the Text or Image (Text to the right, Image to the right).

After creating a button group, you need to move the grouped buttons below of the Button Group and then move them to the right. Like the image below:



Application Hotkeys

Scriptcase allows creating shortcut keys to your applications. You can select a predefined template or create specific actions for an application.

VALUE	DESCRIPTION				
<input checked="" type="checkbox"/> <code>toggleHotkeys();</code>	Define if the application will use hotkeys				
SC_DefaultHotkeys	Select the hotkey template from previously created schemas				
Clear <input type="button" value="+"/>					
<table border="1"><thead><tr><th>ACTION</th><th>KEYBINDING</th></tr></thead><tbody><tr><td colspan="2" style="text-align: center;">No hotkeys configured</td></tr></tbody></table>		ACTION	KEYBINDING	No hotkeys configured	
ACTION	KEYBINDING				
No hotkeys configured					
Clear <input type="button" value="+"/>					

Use hotkeys
Defines if the application uses hotkeys. When you enable this option, the default shortcut keys settings are disabled.

Hotkeys templat
Select the [hotkey template](#) previously created.

Action
Selects the triggered action when pressing the selected key.

Keybinding
Selects the keys responsible for executing the chosen action.

Add "+"
Adds a new action on the keys list.

Clear
It clears the selected hotkeys preference.

PDF Settings

PDF Settings	
ATTRIBUTE	VALUE
PDF Orientation	Portrait
PDF Format	Letter (216 x 279 mm)
Print Type	Color
Generate Bookmarks	<input checked="" type="checkbox"/>
Page Numbers Format	Simple
Page numbering height	Top
Horizontal position of the page numbering	Right
Upper Margin	<input type="text"/>
Bottom Margin	<input type="text"/>
Right Margin	<input type="text"/>
Left Margin	<input type="text"/>
Print Background	<input checked="" type="checkbox"/>
JS execution time	200
Timeout for chart's image creation	150

PDF Orientation

Allows you to set whether to print in Portrait or Landscape orientation.

PDF Format

It allows you to define the PDF paper format (letter, A4, others).

Print Type

It allows you to set whether the print mode as colored or economical with no colors.

Generate PDF directly

Opens the generated the PDF without the need to display an intermediate page with a link to download the file.

PDF Chart depth output selection

It allows the end-user to decide on the fly the depth of the Group By to exports.

Selecting page break per level

It allows the end-user to define the break level to skip to the next page in the generated PDF.

Configurable PDF

It allows the end-user to configure the PDF parameters on the fly.

Generate bookmarks

Generate bookmarks automatically according to the Group By.

Page numbering format

Set the numbering format as Simple (1,2,3,4,5, ...), Full (1 / n, 2 / n, 3 / n, ...) or no numbering.

Page numbering height

It allows you to align the numbering vertically, at the top or bottom of the page.

Horizontal position of page numbering

It allows the horizontal alignment of page numbering (Left, Center, Right).

Print background

Allows you to print background in the PDF file.

JS Runtime

The maximum time, in seconds, of server waiting when running JS.

Timeout for chart's image creation

Set the timeout to wait for the chart image creation in the PDF

WORD Settings

General settings	
ATTRIBUTE	VALUE
Word configurable	<input type="checkbox"/>
Grid Export modules	<input checked="" type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Summary export modules	<input type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Export with password	<input type="checkbox"/>
Select columns	<input type="checkbox"/>
Open WORD Directly	<input type="checkbox"/>

Word configurable

It allows the end-user to configure the export parameters on the fly.

Export Modules of Grid

This option allows the developer to choose which modules are available in the grid export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
---------------------	---

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Summary Export Modules

This option allows the developer to choose which modules are available in the summary export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
---------------------	---

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Export with password

This option allows the developer to set a password to export the file.

Export with password	<input checked="" type="checkbox"/>
Export with password	<input type="text"/>

Password

It allows the developer to set the requested password when exporting the file.

Runtime Example:

PDF settings	
GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Print settings	Color
Page Layout	A4 (210 X 297 mm)
Orientation	Portrait
Other options	<input checked="" type="checkbox"/> Generate Bookmarks <input checked="" type="checkbox"/> Displays the header on all pages <input checked="" type="checkbox"/> Displays the title on all pages
Password

Select columns

It allows the user has the option to choose the columns he wants to export at runtime.

PDF settings	Select Columns
	<input type="checkbox"/> Accountid <input type="checkbox"/> Accountdescription <input type="checkbox"/> Accountorder

Open WORD Directly

Opens the generated WORD file without the need to display an intermediate page with a link to download it.

WORD Settings

WORD Settings		
ATTRIBUTE	VALUE	DESCRIPTION
Print Type	Both	Define the Word print mode.
Open WORD Directly	<input type="checkbox"/>	Open the generated WORD file without creating an intermediary page with a link to it.

Print Type

It allows you to set whether the print mode as colored or economical with no colors(Black and White).

CSV Settings

General settings	
ATTRIBUTE	VALUE
Configurable CSV	<input type="checkbox"/>
Grid Export modules	<input checked="" type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary
Summary export modules	<input type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary
Export with password	<input type="checkbox"/>
Select columns	<input type="checkbox"/>
Open CSV Directly	<input type="checkbox"/>

Configurable CSV

It allows the end-user to configure the CSV export parameters on the fly.

Export Modules of Grid

This option allows the developer to choose which modules are available in the grid export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid	<input type="checkbox"/> Summary	<input checked="" type="checkbox"/> Chart
---------------------	--	----------------------------------	---

You can see the Summary option disabled at running time.

GENERAL			
Select the modules	<input checked="" type="checkbox"/> Grid	<input type="checkbox"/> Summary	<input checked="" type="checkbox"/> Chart

Summary Export Modules

This option allows the developer to choose which modules are available in the summary export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid	<input type="checkbox"/> Summary	<input checked="" type="checkbox"/> Chart
---------------------	--	----------------------------------	---

You can see the Summary option disabled at running time.

GENERAL			
Select the modules	<input checked="" type="checkbox"/> Grid	<input type="checkbox"/> Summary	<input checked="" type="checkbox"/> Chart

Export with password

This option allows the developer to set a password to export the file.

Export with password	<input checked="" type="checkbox"/>
Export with password	<input type="text"/>

Password

It allows the developer to set the requested password when exporting the file.

Runtime Example:

PDF settings	
GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Print settings	Color
Page Layout	A4 (210 X 297 mm)
Orientation	Portrait
Other options	<input checked="" type="checkbox"/> Generate Bookmarks <input checked="" type="checkbox"/> Displays the header on all pages <input checked="" type="checkbox"/> Displays the title on all pages
Password

Select columns

It allows the user has the option to choose the columns he wants to export at runtime.

PDF settings	
Select Columns	
	<input type="checkbox"/> Accountid <input type="checkbox"/> Accountdescription <input type="checkbox"/> Accountorder

Open CSV Directly

Opens the generated CSV file without the need to display an intermediate page with a link to download it.

CSV Settings

CSV Settings	
ATTRIBUTE	VALUE
Lines Separator	CRLF
Columns Separator	:
Text Delimiter	"
Add Label	<input type="checkbox"/>

Line separator

It allows you to define the line separator character, each line representing a record.

Column separator

It allows you to define the column separator character, each column representing a database field.

Text delimiter

It allows you to define the character used to delimit the text of the columns when we have database fields of type String.

Add label

It allows you to define displaying the label of the columns in the first line of the file.

XLS Settings

General settings	
ATTRIBUTE	VALUE
Export with password	<input type="checkbox"/>
Select columns	<input checked="" type="checkbox"/>
Open Excel Directly	<input type="checkbox"/>

Export with password

This option allows the developer to set a password to export the file.

Export with password	<input checked="" type="checkbox"/>
Export with password	<input type="text"/>

Password

It allows the developer to set the requested password when exporting the file.

Runtime Example:

PDF settings	
GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Print settings	Color
Page Layout	A4 (210 X 297 mm)
Orientation	Portrait
Other options	<input checked="" type="checkbox"/> Generate Bookmarks <input checked="" type="checkbox"/> Displays the header on all pages <input checked="" type="checkbox"/> Displays the title on all pages
Password

Select columns

It allows the user has the option to choose the columns he wants to export at runtime.

PDF settings		Select Columns
		Accountid
		Accountdescription
		Accountorder

XLS Configurable

It allows the end-user to configure the XLS export parameters on the fly.

Open XLS Directly

It allows the end-user to configure the XLS export parameters on the fly.

XLS Settings

ATTRIBUTE	VALUE	DESCRIPTION
Excel Configurable	<input checked="" type="checkbox"/>	Allows the user to configure the parameters of creation of Excel during the execution of the application.
Excel settings available to the end-user.		
Grid Export modules	<input checked="" type="checkbox"/>	Defines which modules will be exported in the Grid.
Summary export modules	<input checked="" type="checkbox"/>	
Format	xlsx	Excel format.
Export with totals	<input checked="" type="checkbox"/>	Enables the display of the totals when exporting to Excel.

Export Modules of Grid

This option allows the developer to choose which modules are available in the grid export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/>	Grid
	<input type="checkbox"/>	Summary
	<input checked="" type="checkbox"/>	Chart

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Summary Export Modules

This option allows the developer to choose which modules are available in the summary export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/>	Grid
	<input type="checkbox"/>	Summary
	<input checked="" type="checkbox"/>	Chart

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Format

It allows defining the Excel format of the generated document (xls or xlsx).

Export with totals

Show the totals when exporting the application data to Excel.

XML Settings

General settings	
ATTRIBUTE	VALUE
Configurable XML	<input type="checkbox"/>
Grid Export modules	<input checked="" type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary
Summary export modules	<input type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary
Export with password	<input type="checkbox"/>
Select columns	<input type="checkbox"/>
Open XML Directly	<input type="checkbox"/>

Configurable XML

It allows the end-user to configure the XML export parameters on the fly.

Export Modules of Grid

This option allows the developer to choose which modules are available in the grid export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
---------------------	---

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Summary Export Modules

This option allows the developer to choose which modules are available in the summary export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
---------------------	---

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Export with password

This option allows the developer to set a password to export the file.

Export with password	<input checked="" type="checkbox"/>
Export with password	<input type="text"/>

Password

It allows the developer to set the requested password when exporting the file.

Runtime Example:

PDF settings	
GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Print settings	Color
Page Layout	A4 (210 X 297 mm)
Orientation	Portrait
Other options	<input checked="" type="checkbox"/> Generate Bookmarks <input checked="" type="checkbox"/> Displays the header on all pages <input checked="" type="checkbox"/> Displays the title on all pages
Password

Select columns

It allows the user has the option to choose the columns he wants to export at runtime.

The 'Select Columns' dialog box is shown with a dashed border. It contains a list of columns: Accountid, Accountdescription, and Accountorder. The 'Accountid' column is highlighted, indicating it is selected for export.

Open XML Directly

Opens the generated XML file without the need to display an intermediate page with a link to download it.

XML Settings

ATTRIBUTE	VALUE
Use Label on the Tag	<input type="checkbox"/>
XML Format	<input type="radio"/> Attr <input checked="" type="radio"/> Tag

Use Label on the Tag

It allows using the field labels on Tags of the XML file.

XML Format

Define how to generate the records in the XML file, if it will store the values on attributes or new elements. (Attr or Tag)

Print Settings

ATTRIBUTE	VALUE
Configurable Print HTML	<input type="checkbox"/>
Grid Export modules	<input checked="" type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Summary export modules	<input type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Select columns	<input type="checkbox"/>

Configurable Print HTML

It allows the end-user to configure the HTML Printing parameters on the fly.

Export Modules of Grid

This option allows the developer to choose which modules are available in the grid export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules: Grid, Summary, Chart

You can see the Summary option disabled at running time.

GENERAL
Select the modules: Grid, Summary, Chart

Summary Export Modules

This option allows the developer to choose which modules are available in the summary export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules: Grid, Summary, Chart

You can see the Summary option disabled at running time.

GENERAL
Select the modules: Grid, Summary, Chart

Export with password

This option allows the developer to set a password to export the file.

Export with password

Export with password

Password

It allows the developer to set the requested password when exporting the file.

Runtime Example:

PDF settings

GENERAL

Select the modules Grid Summary Chart

Print settings

Page Layout

Orientation

Other options Generate Bookmarks
 Displays the header on all pages
 Displays the title on all pages

Password

Select columns

It allows the user has the option to choose the columns he wants to export at runtime.

PDF settings **Select Columns**

Print settings ?

ATTRIBUTE	VALUE	DESCRIPTION
Print Type	<input type="text" value="Both"/>	Define the print mode.
HTML Chart depth output selection	<input checked="" type="checkbox"/>	Allows the user to decide to what depth of the group by charts will be generated in the HTML.
Print Background	<input checked="" type="checkbox"/>	Display background in a HTML form.

Print Mode

It allows choosing the content to print. (Current Page or Full Report)

Print Background

Allows displaying the background in an HTML form.

RTF Settings

General settings

ATTRIBUTE	VALUE
Select columns	<input type="checkbox"/>
Open RTF Directly	<input type="checkbox"/>

Select columns

This option allows the end-user to select which fields he wants to export to the file.

PDF settings **Select Columns**

Open RTF Directly

Opens the generated RTF file without the need to display an intermediate page with a link to download it.

SQL Settings

This interface allows configuring the related database settings, such as the SQL statement, the used database connection, case sensitive, and others.

ATTRIBUTE	VALUE	DESCRIPTION
SQL Select Statement	<pre>SELECT customerid, companyname, contactname, contacttitle, birthdate, country, regionid, stateid, city,</pre>	
Limit	<input type="text"/>	It sets the number of records to be retrieved from the SQL statement.
SQL Preparation	<input type="text"/>	
Connection	<input type="text" value="conn_example"/>	Connection name to access the database.
Use Customized Message	<input type="checkbox"/>	Use a customized error message when the application has no records.
No Records Message	<input type="text"/>	When the application has no records, it will display this customized text.
Font	<input type="text" value="Aa"/>	Font face of the error message.
Font Size	<input type="text" value="12"/>	Font size of the error message.
Font Color	<input type="text" value="#000000"/>	Font color of the error message.
Variable for Table	<input type="text"/>	Variable name used for replacing the table name. Please indicate the name of the table that will be replaced by the variable value.
Fields Variables	<div style="border: 1px solid #ccc; padding: 5px;"> <p>Variable</p> <input type="text"/> <p>customerid</p> </div>	Variables for substitution of the field names on the application. For each dynamically determined field, inform the name of the variable and the field that will be substituted.
Case Sensitive	<input checked="" type="checkbox"/>	Use case sensitive.

Grid SQL configuration

SQL Select Statement

It allows you to define the primary SQL of the application. You can edit this SQL to add or delete fields.

Limit

Lets you limit the display in the number of records retrieved by SQL query.

SQL Preparation

You can enter SQL commands or procedure names to execute them before the primary SQL of the application.

Connection

It allows defining the database connection of the application. You can change the connection to another one that has the same table.

Use Customized Message

Lets you define to display the "no records" message or not.

No Records Message

Lets you set the message when the application has no records.

Font

This option is available when using the "Use Customized Message" option. It lets you set the font for the message.

Font Size

This option is available when using the "Use Customized Message" option. It lets you to set the font size.

Font Color

This option is available when using the "Use Customized Message" option. It lets you to set the font color.

Variable for Table

It allows to use a variable to change a part of the string containing the table name.

Fill the first input with the name of the variable (replace to). The second input you should fill with the part of the name of the table you want to replace (replace from).

Fields Variables

It allows to use a variable to change a part of the string containing the field name.

Fill the first input with the name of the variable (replace to). The second input you should fill with the part of the name of the field you want to replace (replace from).

Case sensitive

It defines if the database connection uses case sensitive or not.

Security ?		
ATTRIBUTE	VALUE	DESCRIPTION
Use Security	<input type="checkbox"/>	Application needs validation to run. Read about the sc_apl_status macro.
Url output of the security	<input type="text"/>	Uri to be redirected if the user does not have access to this application.
Use Password	<input type="checkbox"/>	Enter password required to run the application.
Request password just once	<input type="checkbox"/>	Request password just once per session without asking for password every time you open the application in the same session.
Enable direct call by URL	<input checked="" type="checkbox"/>	Allow an application to be called by typing the URL directly in the browsers.

Application Security Configuration

Use Security

Turns the application accessible just when using the macro sc_apl_status or using the security module. When it is enabled, the application displays a message "User not Authorized" if the user tries to access it directly.

Url output of the security

It redirects the application to another URL after displays the warning "User not Authorized".

Use Password

Turns the application accessible just by using a password.

Request password just once

Request the password once per session, not needing to inform the password every time you access the application.

Enable direct call by URL

Allows to call the application directly from the URL in the browsers.

Allows calling the application directly from the URL in the browsers. In the development environment, you can disable those options for testing, it avoids requiring the login every time you run the application. To disable the security and password in development, you must access Options > My ScriptCase and uncheck the options: Enable Use of Security, Enable Use of Password.

This interface allows you to define a Log schema to the app. The Log schema tracks what the end-users are doing in the app. Those are the events it can tracks: Access, Insert, Update, and delete.

To learn how to create a Log schema [see here](#).

The screenshot shows a configuration window titled "LOG". It features a table with two columns: "ATTRIBUTE" and "VALUE". The "Schema" attribute has a dropdown menu currently set to "log". Below the table, there are two empty list boxes: "Events" on the left and "Access" on the right. Between these two boxes are two blue buttons: "Add >>" and "Remove <<".

Application Log configuration

Schema Select an existing log schema for the application.

Events You can select just specific events to save in the log.

1. [Home](#)
2. Applications

General Overview

The applications can have two types of fields.

Table Fields

Columns of the table connected to the application and their data types.

Example:

In a table with the following structure:

```
CREATE TABLE `orders` (
  `orderid` int(11) NOT NULL,
  `customerid` varchar(5) NULL DEFAULT NULL,
  `employeeid` int(11) NULL DEFAULT NULL,
  `orderdate` date NULL DEFAULT NULL,
  `requireddate` date NULL DEFAULT NULL,
  PRIMARY KEY (`orderid`) USING BTREE
);
```

These are the listed fields on left menu **Fields**:

Virtual Fields

Fields created inside application to assist in the development process.

How to create a virtual field

1. Inside an application, access the left menu **Fields** and click on **New Field** option.
2. Choose how many fields do you want to create:

1. Create the fields and define type, name and label for each one:

- **Type:** Data type of the field.
- **Name:** Internal name of the field. Used to identify the field on events and Scriptcase interface.
- **Label:** Title of the field displayed on the application.

Scriptcase **doesn't insert** virtual fields on the table after its creation.

Types of fields by applications

Application Table Fields Virtual Fields

Grid	<input type="text"/>	<input type="text"/>
Procedure	<input type="text"/>	<input type="text"/>
Chart	<input type="text"/>	<input type="text"/>
Form	<input type="text"/>	<input type="text"/>
Control	<input type="text"/>	<input type="text"/>
Search	<input type="text"/>	<input type="text"/>
PDF Report	<input type="text"/>	<input type="text"/>
Calendar	<input type="text"/>	<input type="text"/>

The applications **Menu**, **Tree Menu**, **Dashboard** and **Blank** were not listed because they haven't manage fields.

1. [Home](#)
2. [Applications](#)
3. Text

Text

General Settings

This type of field allows the developer to create quickly inputs to insert and update data, where the final user can inform its data to be allocated in its database.

Type of Data

Defines the type of the field in the application. In this case we should select **Text**.

Label

Defines the title that will be displayed in the field when the application is executed. The terminology used in the interface is fundamental to the system have a good usability, who should use names and familiar terms to the final user of the application, instead of using terms from the system.

For example, this text field that has the name **customerid**, the client would have a much better understanding of the functionality of the field when we define the label as **Customer Name**.

Not only a fixed text, the **Label** attribute allows the use of langs to define the field's title, making it possible to use your application in a multi language project.

SQL Type

Informs the type of the field in the database.

Lookup Settings {id-02}

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets `{}`. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
- **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
- **Summary** : Applies the lookup only for the summary (HTML and PDF).
- **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.

- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Martial Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1			Sports
2			Culture
4			Pleasure
8			Reading
16			Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1			Sports
2			Culture
4			Pleasure
8			Reading
16			Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute	Value	Lookup	Description
1			Sports
2			Culture
4			Pleasure
8			Reading
16			Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

1. [Home](#)
2. [Applications](#)
3. Multiple Lines Text

Multiple Lines Text

General Settings

Multiple Lines Text field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Multiple Lines Text , you can inform a Text value to the field in multiple lines.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **SQL Type** : Informs the data type of field in the database.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Martial Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1

Label	Value	Start	Size
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

1. [Home](#)
2. [Applications](#)
3. Integer

Integer

General Settings

Data type:

It sets the application field type. When the field is set as Number, it is permissible to define formatting rules of integers.

Label:

It sets the title that will be displayed in the field when you run the application. The terminology used in the interface is fundamental for your system has a good usability. In this case, we recommend you to use names and terms familiar to the end user of the application, instead of using terms originated in the system.

Field Mask Input:

It sets the mask input for the field. There are two types of mask as described in the table below:

Character Description

X	It is overridden by any characters returned by the database. Filled is required and the value will be completed with leading zeros when there are fewer characters than the size of the mask.
Z	It is overridden by any characters returned by the database. Its filling is optional and when there are fewer characters than the mask size, nothing will be done about the characters that are missing. In addition the leading zeros will be suppressed. When used in conjunction with the X necessarily the use of this character is to the left of the mask.
9	It represents any numeric character (from 0-9)
A	It represents an alpha numeric character (A-Z,a-z)
*	It represents any alpha-numeric character (A-Z,a-z,0-9)entered by the user.

Mask Input Examples:

Field	Mask Input	Typed Value	Formatted Value
Phone Number	+99 99 9999 - 9999	123456789012	+12 34 5678 - 9012
Phone Number	(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
Phone Number	(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
Phone Number	(zz) xxxx - xxxx	12345678	() 1234 - 5678
Software Key (Letters only)	AAAA-AAAA-AAAA-AAAA	QWERTYUIASDFGHJK	QWER-TYUI-ASDF-GHJK
Software Key (Letters and numbers)	xxxx-xxxx-xxxx-xxxx	Q1W2E3R4T5Y6U7I8	Q1W2-E3R4-T5Y6-U7I8
License Plate	AAA - 9999	QWE1234	QWE - 1234
Scriptcase Serial Key	A999A999A999-**	D111H111G111DG2P	D111H111G111-DG2P
Multiple masks (Phone Number)	9999-9999;(99)9999-9999; 9999 999 9999		+99 99 9999-9999

Repeat value:

This option when enabled will allows you to repeat the field value if it is equal to the value of the previous record in the database.

Example:

Tipo SQL:

Reports the SQL type of the field as it is configured in the database.

Values Format

Use regional settings:

It allows you to apply regional settings to the number fields formatting. When this option is not selected it will be displayed the attributes group, minus sign and negative number format.

Digit grouping:

It allows you to define if the field displays the digit grouping separator.

Maximum Size:

It allows you to define the maximum size that will be used in the field of the application.

Minimum Value:

It allows you to define the minimum value that will be used in the field of the application.

Maximum Value:

It allows you to define the maximum value that will be used in the field of the application.

Grouping:

It allows you to define which character is used to separate thousands. This option is only available when the option to use regional settings is disabled.

Negative sign:

It allows you to define which character will be used for the display of negative numbers. This option is only available when the option to use regional settings is disabled.

Negative number format:

It allows you to define the placement of the negative sign in relation to value. This option is only available when the option to use regional settings is disabled.

Accept:

It allows to define if the field accept only negative values, positive-only or both.

Display Calculator:

This option when enabled will display a calculator to assist the end user to perform calculations.

Use Spin:

This option applies a Spin component to the number field. So the end user can increase or decrease the numbers using it. You will also be able to set a minor or major range for the spin.

Smaller increment for the Spin::

Sets the minimum range of the Spin component.

Greater increment for the Spin:

Sets the maximum range of the Spin.

Lookup Settings

The Lookup is a way used to provide the end user a list of values description/name where the items can be selected and used in the application. For example, for a field sex that in the database is represented with M or F values can be presented as male or female dynamically with the lookup. So you will be able to retrieve these values from a database table or manually, if the reference table does not exist.

Example:

Lookup Methods

There are two lookup methods available:

Manual: To use this method it is necessary to inform the selection conditions manually to the Lookup.

Automatic: To use this method it is necessary to use a lookup query from the database to access the reference table.

Editing-automatic lookup:

Select Command:

It sets the SQL command that will retrieve the value to display inside the Grid field. The command must have the following format

```
SELECT field_to_be_displayed FROM table WHERE key_field = {grid_field}
```

The Grid field must be always referenced between curly braces {}, So at run time, the field braces will be replaced by its value in the table.

Multiple Options:

When the Select command informed return multiple values, the option must be selected.

Delimiter:

It sets the tab of values returned by the Select command, this option should be filled when the "Yes" option is checked in the multiple attribute options.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

Use a specific connection:

This option allows you to select another existing connection in the project, other than the current connection, to select the values for the lookup.

Edit Lookup - Manual:

Lookup used to modify the display of the field with predefined values.

Lookup Types

It sets the operating mode of the lookup can be: single value, multiple values (delimiter), multiple values (position) and multiple values (binary).

Single value: it will be displayed only one value for the field. There is a direct replacement of the value stored in the Bank for a label defined in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value of "M" will be replaced by "Male".

Multiple Values (Delimiter):

By using this option it will be displayed multiple values for the selected field. The values must be separated by a delimiter to be informed. A replacement of parts from a string stored in the field, separated by a delimiter for values contained in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value S;M will be replaced by Sports and Movies.

Delimiter: Character used to separate the values within the same string.

Multiple Values (Position)

Allows you to recover information stored in a single string of the selected field. In order for this information to be retrieved must be informed, in addition to the label, the starting position and the number of bytes that each information occupies within the string.

As an example we will use a string to store Sex, Marital status and Hobby respectively. Sex occupies one byte, Status one byte and Hobby two bytes .

For this purpose we define the list as:

Label	Value	Start	Size
Male	M	1	1
Female	F	1	1
Married	M	2	1
Single	S	2	1
Study	ST	3	2
Sportes	SP	3	2
Reading	LE	3	2

Example: the string MMST would be presented in the query as: Male Married Study

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value M will be replaced by Male.

Start: : Starting position in the string where the information is recorded. The first position is always 1.

Size : Number of bytes that the information occupies in the string.

Multiple values (binary):

It allows you to retrieve several information stored in decimal form of the selected field.

As examples we will use the following list (although not informed in the interface of inclusion of values in the list, each item has a value assigned automatically according to the order):

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

To display the data, a decomposition of the decimal number stored in the bank is performed. For example, numbers 11 and 12 (written in the database) would be decomposed as follows:

$11 = 1 + 2 + 8 = (\text{Sports} - \text{Culture} - \text{Reading})$

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

$12 = 4 + 8 = (\text{Leisure} - \text{Reading})$

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading

Assigned value	Description in Lookup

Label:

Text that will be displayed in the Grid.

Include Button:

Includes in the list the values populated in the Label and value fields.

Alter Button:

It changes the attributes of the selected item.

Delete Button:

Deletes the selected item.

Clean Button:

It cleans the fields

Save Button:

Allows the user to save the entire list, for later use using the Load Lookup definition option.

Load Lookup definition:

Allows you to load a predefined value list for use as a lookup.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Decimal

Decimal

General Settings

Decimal field configuration Interface.

- **Data Type** : Select the type of field for the application. When it is defined as a Decimal you can define the format of a decimal number.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **SQL Type** : Informs the data type of field in the database.

Values Format

Use regional settings:

It allows you to apply regional settings to the number fields formatting. When this option is not selected it will be displayed the attributes group, minus sign and negative number format.

Digit grouping:

It allows you to define if the field displays the digit grouping separator.

Maximum Size:

It allows you to define the maximum size that will be used in the field of the application.

Minimum Value:

It allows you to define the minimum value that will be used in the field of the application.

Maximum Value:

It allows you to define the maximum value that will be used in the field of the application.

Grouping:

It allows you to define which character is used to separate thousands. This option is only available when the option to use regional settings is disabled.

Negative sign:

It allows you to define which character will be used for the display of negative numbers. This option is only available when the option to use regional settings is disabled.

Negative number format:

It allows you to define the placement of the negative sign in relation to value. This option is only available when the option to use regional settings is disabled.

Accept:

It allows to define if the field accept only negative values, positive-only or both.

Display Calculator:

This option when enabled will display a calculator to assist the end user to perform calculations.

Use Spin:

This option applies a Spin component to the number field. So the end user can increase or decrease the numbers using it. You will also be able to set a minor or major range for the spin.

Smaller increment for the Spin::

Sets the minimum range of the Spin component.

Greater increment for the Spin:

Sets the maximum range of the Spin.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced by the curly brackets {}. While running the application, the field within the curly brackets is replaced by its value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value Lookup Description

1	Sports
2	Culture

Attribute Value Lookup Description

8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports, Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value**: When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Percent

Percent

General Settings

Percentage field configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Percentage, it is allowed to define the format of a percentage.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **SQL Type** : Informs the data type of field in the database.

Values Format

Use regional settings:

It allows you to apply regional settings to the number fields formatting. When this option is not selected it will be displayed the attributes group, minus sign and negative number format.

Digit grouping:

It allows you to define if the field displays the digit grouping separator.

Maximum Size:

It allows you to define the maximum size that will be used in the field of the application.

Minimum Value:

It allows you to define the minimum value that will be used in the field of the application.

Maximum Value:

It allows you to define the maximum value that will be used in the field of the application.

Grouping:

It allows you to define which character is used to separate thousands. This option is only available when the option to use regional settings is disabled.

Negative sign:

It allows you to define which character will be used for the display of negative numbers. This option is only available when the option to use regional settings is disabled.

Negative number format:

It allows you to define the placement of the negative sign in relation to value. This option is only available when the option to use regional settings is disabled.

Accept:

It allows to define if the field accept only negative values, positive-only or both.

Display Calculator:

This option when enabled will display a calculator to assist the end user to perform calculations.

Use Spin:

This option applies a Spin component to the number field. So the end user can increase or decrease the numbers using it. You will also be able to set a minor or major range for the spin.

Smaller increment for the Spin::

Sets the minimum range of the Spin component.

Greater increment for the Spin:

Sets the maximum range of the Spin.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced by the curly brackets {}. While running the application, the field within the curly brackets is replaced by its value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1			Sports
2			Culture
4			Pleasure
8			Reading
16			Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value Lookup Description

1	Sports
2	Culture

Attribute Value	Lookup Description
8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports, Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value	Lookup Description
1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value**: When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Currency

Currency

General Settings

Currency field configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Decimal, it is allowed to define the format of a currency number.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **SQL Type** : Informs the data type of field in the database.

Values Format

Use regional settings:

It allows you to apply regional settings to the number fields formatting. When this option is not selected it will be displayed the attributes group, minus sign and negative number format.

Digit grouping:

It allows you to define if the field displays the digit grouping separator.

Maximum Size:

It allows you to define the maximum size that will be used in the field of the application.

Minimum Value:

It allows you to define the minimum value that will be used in the field of the application.

Maximum Value:

It allows you to define the maximum value that will be used in the field of the application.

Grouping:

It allows you to define which character is used to separate thousands. This option is only available when the option to use regional settings is disabled.

Negative sign:

It allows you to define which character will be used for the display of negative numbers. This option is only available when the option to use regional settings is disabled.

Negative number format:

It allows you to define the placement of the negative sign in relation to value. This option is only available when the option to use regional settings is disabled.

Accept:

It allows to define if the field accept only negative values, positive-only or both.

Display Calculator:

This option when enabled will display a calculator to assist the end user to perform calculations.

Use Spin:

This option applies a Spin component to the number field. So the end user can increase or decrease the numbers using it. You will also be able to set a minor or major range for the spin.

Smaller increment for the Spin::

Sets the minimum range of the Spin component.

Greater increment for the Spin:

Sets the maximum range of the Spin.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced by the curly brackets {}. While running the application, the field within the curly brackets is replaced by its value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
- **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	

Attribute Value Lookup Description

8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports, Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Percent(Calculated)

Percent(Calculated)

General Settings

Percentage field configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Percentage, it is allowed to define the format of a percentage.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **SQL Type** : Informs the data type of field in the database.

Values Format

Use regional settings:

It allows you to apply regional settings to the number fields formatting. When this option is not selected it will be displayed the attributes group, minus sign and negative number format.

Digit grouping:

It allows you to define if the field displays the digit grouping separator.

Maximum Size:

It allows you to define the maximum size that will be used in the field of the application.

Minimum Value:

It allows you to define the minimum value that will be used in the field of the application.

Maximum Value:

It allows you to define the maximum value that will be used in the field of the application.

Grouping:

It allows you to define which character is used to separate thousands. This option is only available when the option to use regional settings is disabled.

Negative sign:

It allows you to define which character will be used for the display of negative numbers. This option is only available when the option to use regional settings is disabled.

Negative number format:

It allows you to define the placement of the negative sign in relation to value. This option is only available when the option to use regional settings is disabled.

Accept:

It allows to define if the field accept only negative values, positive-only or both.

Display Calculator:

This option when enabled will display a calculator to assist the end user to perform calculations.

Use Spin:

This option applies a Spin component to the number field. So the end user can increase or decrease the numbers using it. You will also be able to set a minor or major range for the spin.

Smaller increment for the Spin::

Sets the minimum range of the Spin component.

Greater increment for the Spin:

Sets the maximum range of the Spin.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced by the curly brackets {}. While running the application, the field within the curly brackets is replaced by its value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
- **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1			Sports
2			Culture
4			Pleasure
8			Reading
16			Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value Lookup Description

1	Sports
2	Culture

Attribute Value Lookup Description

8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports, Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Date

Date

General Settings

Date field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Date and Time, it is allowed to define the format of the date.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Date format** : Format the dates used for the Group By.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Use regional settings:

It allows you to apply regional settings to the number fields formatting. When this option is not selected it will be displayed the attributes group, minus sign and negative number format.

Digit grouping:

It allows you to define if the field displays the digit grouping separator.

Maximum Size:

It allows you to define the maximum size that will be used in the field of the application.

Minimum Value:

It allows you to define the minimum value that will be used in the field of the application.

Maximum Value:

It allows you to define the maximum value that will be used in the field of the application.

Grouping:

It allows you to define which character is used to separate thousands. This option is only available when the option to use regional settings is disabled.

Negative sign:

It allows you to define which character will be used for the display of negative numbers. This option is only available when the option to use regional settings is disabled.

Negative number format:

It allows you to define the placement of the negative sign in relation to value. This option is only available when the option to use regional settings is disabled.

Accept:

It allows to define if the field accept only negative values, positive-only or both.

Display Calculator:

This option when enabled will display a calculator to assist the end user to perform calculations.

Use Spin:

This option applies a Spin component to the number field. So the end user can increase or decrease the numbers using it. You will also be able to set a minor or major range for the spin.

Smaller increment for the Spin::

Sets the minimum range of the Spin component.

Greater increment for the Spin:

Sets the maximum range of the Spin.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Time

Time

General Settings

Time field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Time, it is allowed to define the format of the time.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **SQL Type** : Informs the data type of field in the database.

Values Format

Use regional settings:

It allows you to apply regional settings to the number fields formatting. When this option is not selected it will be displayed the attributes group, minus sign and negative number format.

Digit grouping:

It allows you to define if the field displays the digit grouping separator.

Maximum Size:

It allows you to define the maximum size that will be used in the field of the application.

Minimum Value:

It allows you to define the minimum value that will be used in the field of the application.

Maximum Value:

It allows you to define the maximum value that will be used in the field of the application.

Grouping:

It allows you to define which character is used to separate thousands. This option is only available when the option to use regional settings is disabled.

Negative sign:

It allows you to define which character will be used for the display of negative numbers. This option is only available when the option to use regional settings is disabled.

Negative number format:

It allows you to define the placement of the negative sign in relation to value. This option is only available when the option to use regional settings is disabled.

Accept:

It allows to define if the field accept only negative values, positive-only or both.

Display Calculator:

This option when enabled will display a calculator to assist the end user to perform calculations.

Use Spin:

This option applies a Spin component to the number field. So the end user can increase or decrease the numbers using it. You will also be able to set a minor or major range for the spin.

Smaller increment for the Spin::

Sets the minimum range of the Spin component.

Greater increment for the Spin:

Sets the maximum range of the Spin.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Date and Time

Date and Time

General Settings

Date and Time field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Date and Time you can define the format of the date.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Date format** : Format the dates used for the Group By.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Use regional settings:

It allows you to apply regional settings to the number fields formatting. When this option is not selected it will be displayed the attributes group, minus sign and negative number format.

Digit grouping:

It allows you to define if the field displays the digit grouping separator.

Maximum Size:

It allows you to define the maximum size that will be used in the field of the application.

Minimum Value:

It allows you to define the minimum value that will be used in the field of the application.

Maximum Value:

It allows you to define the maximum value that will be used in the field of the application.

Grouping:

It allows you to define which character is used to separate thousands. This option is only available when the option to use regional settings is disabled.

Negative sign:

It allows you to define which character will be used for the display of negative numbers. This option is only available when the option to use regional settings is disabled.

Negative number format:

It allows you to define the placement of the negative sign in relation to value. This option is only available when the option to use regional settings is disabled.

Accept:

It allows to define if the field accept only negative values, positive-only or both.

Display Calculator:

This option when enabled will display a calculator to assist the end user to perform calculations.

Use Spin:

This option applies a Spin component to the number field. So the end user can increase or decrease the numbers using it. You will also be able to set a minor or major range for the spin.

Smaller increment for the Spin::

Sets the minimum range of the Spin component.

Greater increment for the Spin:

Sets the maximum range of the Spin.

This module lets you editing display options for the application. You can apply visualization themes, organize blocks, define values, and the display format of the Header and Footer.

Layout Settings

ScriptCase creates an app by using default values defined per project, but you can change those values for a specific application on this interface.

LAYOUT SETTINGS

Header Template	<input type="text" value="Flat"/>	Template name used for the application header.
Footer Template	<input type="text" value="Default"/>	Template name used for the application footer.
Button	<input type="text"/>	Use different buttons than what was defined in the color scheme.
Themes	<input type="text" value="Sc9_Rhino"/>	Use different themes from the one defined for the color scheme

Header

|< < > >| xxyyzz xxxxx yyyyy ▾

Block 1

Name

Type Male Female

Address*

Groups* Male Female

Countries

Address

Photos

Drag & Drop files here

Image1.png ✓
Image2.png ✗

Captcha

Application Layout Settings

Header Template

Allows choosing the look of the application Header.

Footer Template

Allows choosing the look of the application Footer.

Button

Allows choosing the button theme for the application.

Themes

Choose one of the selected themes in the project properties. It defines the look of the application, like colors, fonts, and others.

Header & Footer

Header

Here we can define if we want to display the header or not, the application title, and fill the header variables. The header looks according to the selected template in the Layout settings interface.

HEADER	
ATTRIBUTE	VALUE
Display Header	<input checked="" type="checkbox"/>
Insertion Title	{lang_othr_frmj_tit} - orders
Update Title	{lang_othr_frmu_tit} - orders
Header Variables	
SC_HEADER	Title
SC_VALUE	Date

Application Header Configuration

Display Header


This option determines if we want to display the header or not.

Title

Allows informing the title of the application.

Header Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **itle**: It displays the value of "**Application Title**" in the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Depending on the Application, you may have more than one title option.

Footer

FOOTER	
ATTRIBUTE	VALUE
Display Footer	<input type="checkbox"/>
Footer Variables	
NM_FOOTER_DATA	
NM_FOOTER_LOGO	
NM_FOOTER_TITLE	

Application Footer Settings


This page may change depending on the footer template chosen in the Layout Settings.

Display Footer

This option determines if we want to display the footer or not.

Footer Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Scriptcase uses the events to enable the developer to customize the application code. Using the events you can program custom actions at a specific execution time (e.g. After a record is inserted, when it is loaded, upon submitting a form, ...) and for a specific application type. In the event areas you can use global and local variables, JavaScript, CSS, HTML, PHP codes and also Scriptcase macros.

onApplicationInit

This event occurs only once, when the application is loaded and before the application executes the SQL. It can be used for data processing or checking variables.

Example: `if ([glo_var_dept] != 'financial'){ sc_redir(app_x.php); }`

OnScriptInit

This event occurs when the application scripts are started. It runs before the OnLoad events(when present), however, it will always be executed when running, reloading or navigating the application.

onHeader

This event runs when the header of the application is loaded.

onFooter

This event runs when the footer of the application is loaded.

Settings

With this interface, you can define general options of the Search Form.

ATTRIBUTE	VALUE
Search Criteria	AND ▼
Display Condition	<input type="checkbox"/>
Use auto-complete in the fields.	Defined in the field ▼

Search configuration Interface.

Search Criteria

Allows to select the logical operator **AND** or **OR** to define the criteria of the search;

Display Condition

Gets the condition of the search available for the user to choose one. He can select "AND" or "OR" in a Combobox.

Use auto-complete in the fields

Automatically turns the field into an autocomplete according to the existing values in the database. If the user chooses **Yes**, the autocomplete will enable automatically in all inputs that contain a relationship. If the user decides **No**, so no autocompletes will be displayed. Otherwise, the option selected is **Defined in the field** it'll keep the settings for each field individually.

Filter Conditions

With this interface, you can configure the conditions available for each field of the Search form.

ATTRIBUTE	VALUE	DESCRIPTION
Check the field that is part of the search and select in the right combo-box an option to enable or disable a search option.		
customerid	*Contains	
companyname	*Not Contains	
contactname	*Equal to	
contacttitle	*Empty	
birthdate	Different	
country	Beginning with	
regionid	Greater than	
stateid	Greater equal	
city	Less than	
address	Less equal	
postalcode	Between two values	
phone	.in	

On/Off
All
None

Search configuration Interface.

We can see the fields list on the left combo. On the right, the list of options for filtering the selected field. To select an option, click on one of them (Equal to, Beginning with, Contains, etc.) and then the button On/Off. The arrows, on the right, allows altering the order of the fields.

For the Date type fields, you can define special conditions for the search, accessing the field configurations, and editing the Special Conditions Settings.

Below the list are the buttons to enable the selected options:

- **On/Off:** Enables or disables the field or the option chosen.
- **All:** Marks all fields or options.
- **None:** Unmarks all the fields or options.

Advanced Search

Settings

Through these attributes you can define some general Search settings, such as alignment, width, search conditions, etc.

ATTRIBUTE	VALUE	DESCRIPTION
Margins	<input type="text"/> Up <input type="text"/> Down <input type="text"/> Right <input type="text"/> Left	Set the margin.
Keep Values	<input checked="" type="checkbox"/>	Maintain the last search values when accessing the search page again.
Keep columns and sorting	<input checked="" type="checkbox"/>	Keeps last columns and order selection
Use Enter to	<input type="text"/>	Use the "Enter" key to move from the current field to the next field.
Display tags	<input checked="" type="checkbox"/>	It displays used filters as tags inside the Grid.
Display after filtered	<input type="checkbox"/>	Displays the tags line only after it has been activated within the advanced search module.
Unify results	<input type="text" value="15"/>	Amount of characters to unify the result of tags. Recommended option for use when the field is multiple selection.
Treeview in the tags	<input checked="" type="checkbox"/>	This option enables the use of treeview in the tags, and it can hide the tags and show a full description as text.
Initial status of the treeview	<input type="text" value="Start open"/>	This option sets whether the initial status will be open, with the tags, or closed, showing the full text description.

Margins

Defines the position of the margins of the Search Form.

Keep Values

It keeps the searched values when the user returns to the search form.

Keep Columns and Order Selection

Set it to preserve the selected columns and sorting for each search, if they went changed by the user through the toolbar options.

Use Enter to

It allows you to define the action that the Enter Key has on the Search form. **Tabulate** enables you to navigate between fields, and **Submit** performs the search(activates the Search button).

Display Tags

Allows displaying as tags, the searches used for the Grid.

Display after filtering

Display tags only after performing an advanced search. If disabled, it will always display a tag, regardless of the advanced search.

Unify results

Sets the chars limit to group the result of the tags. This option should be used when the field type is multiple-select.

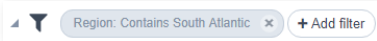
Treeview in the Tags

Sets the use of Treeview for tags.

Initial status of the Treeview

It sets the initial state of the Treeview. (If the app is using Treeview for tags)

Start open

It displays the tags. 

Start close

It displays the full description of the tags as text. 

Select Fields

SEARCH FIELDS ?

Select the fields that will be displayed.

customerid
companyname
contactname
contacttitle
birthdate
country
regionid
stateid
city
address
postalcode
phone
fax
cityid
creditlimit
cardtype
cardnumber
notes

⌘

⌵

⌴

⌘

Pag1
grid_customers
customerid
companyname
contactname
contacttitle
birthdate
country
regionid
stateid
city
address
postalcode
phone
fax
cityid
creditlimit
cardtype
cardnumber
notes

Required

This interface has the options for you to select and set the Search form required fields.

Within the generated application will be displayed a marker(*) next to the field and an error message will also pop up if there's no value assigned to the mandatory fields. You will be able to set some options for the market using the options below.

- **Marker position** : Marker's position relative to the field (options are right, left or Do not display).
- **Display message** : Displays whether or not the validation error message.

Toolbar

Desktop

The Search toolbar is divided in two parts: Top and Bottom, in a way that is possible to define the buttons that will be displayed in both bars. The selection of buttons in the top and bottom toolbar works independently, allowing the buttons to be displayed in both bars at the same time.

Toolbar Interface.

Navigation: Groups the options relative to the navigation buttons that can be displayed in the application.

- **Search:** Execute the search.
- **Clean:** Clean the all the search fields.
- **Edit:** Enable the **Save Tag** option.
- **Exit:** Exit the application.

Others: Groups a diversity of options relative to the application.

- **Languages:** Displays a combobox with the names available, defined in the project properties.
- **Themes:** Displays a combobox with the themes available, defined in the project properties.
- **HelpCase:** Displays a button to redirect to the help page.

Separator

- : Displays a line separating the buttons, when used the Group Buttons.

Use in-line buttons: Allows the alignment of the filter buttons next to the fields.

- **__ Inline buttons __:** Allows you to select which buttons will be displayed next to the field, and you can sort them according to your wishes. This option is available by enabling **Use Online Buttons** in the button settings.

Button Settings

BUTTON SETTINGS			
Button	Label	Hint	Shortcut key
Search			
Clean			
Edit			
Exit			

Button Settings Interface.

- **Hotkey:** Allows you to set keyboard hotkeys to a button.
- **Use Inline Buttons:** Allows alignment of the filter buttons next to the fields.
- **Position of the in-line buttons:** Sets the positioning of the buttons to the right or left of the fields.
- **Column Quantity:** Sets the number of display columns of the buttons, allowing you to configure whether they will be displayed side-by-side or distributed in columns.

Options

OPTIONS	
ATTRIBUTE	VALUE
Buttons Position (top)	Center ▾
Buttons Position (bottom)	Center ▾

Options Interface.

- **Button Position(Top/Bottom)** Positioning the buttons of the toolbar Top/Bottom.

Save Search

This feature allows the end-user to save his searches in a profile. You can create some rules, like to save the searches by user login.

SAVE CRITERIA	
<input checked="" type="checkbox"/> Enable Public	
Title	{lang_srch_public}
<input checked="" type="checkbox"/> Use Rules	
	<div style="border: 1px solid red; padding: 2px; display: inline-block;">New rule</div> <div style="margin-left: 10px;"> <div style="border: 1px solid blue; padding: 2px; display: inline-block;">Edit</div> <div style="margin-left: 10px; border: 1px solid blue; padding: 2px; display: inline-block;">Delete</div> </div>

Save Filter Interface.

SAVE CRITERIA	
Name	{var_login}
Variable name	
Label	
	<div style="border: 1px solid blue; padding: 2px; display: inline-block;">Add</div> <div style="margin-left: 10px; border: 1px solid blue; padding: 2px; display: inline-block;">Delete</div>
Finish	Cancel

Save Filter Interface.

Events

Scriptcase uses the events to enable the developer to customize the application code. Using the events you can program custom actions at a specific execution time (e.g. After a record is inserted, when it is loaded, upon submitting a Form, ...) and for a specific application type. In the event areas you can use global and local variables, JavaScript, CSS, HTML, PHP codes and also Scriptcase macros.

onScriptInit

This event occurs when the application scripts are started. It runs before the OnLoad events(when present), however, it will always be executed when running, reloading or navigating the application.

onRefresh

This event runs when the refresh option of the application is enabled.

onSave

This event runs when the application saves the record.

onValidate

This event runs when validating the information of the fields, when submitting the form.

Layout

This module lets you editing display options for the application. You can apply visualization themes, organize blocks, define values, and the display format of the Header and Footer.

Blocks

Blocks are “containers” where you can position the application fields of Forms, Controls, or Slide Grids.

Scriptcase creates applications with one block by default. You can add more blocks as you wish, to organize it in the best way.

See below, the Columns Organization, and where you can define the position of the next block: beside or below the current one.

Block		Title	Label			Fields		Organization				
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse		
Pag1												
			form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Pag2												
			Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Blocks not shown												

Application Block configuration

On the left side of each block, there are two icons, the first one to edit the information of the block and the second one to delete the block.

Organizing The Position Of The Blocks

See below how to modify the display order of the Blocks in one Page.

Click and drag the block that you desire to modify to its new position.

Block		Title	Label			Fields		Organization				
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse		
Pag1												
			form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Pag2												
			Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Blocks not shown												

Application Block Display configuration

- See how to remove a block from the display

Click on the block desired and drag it to the item “Blocks not Shown”. This way, you can also drag the block to another page if desired. See the images below.

Block		Title	Label			Fields		Organization				
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse		
Pag1												
			form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Pag2												
			Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Blocks not shown												
			Block 3	Block 3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open

Application Block Display configuration

Block		Title	Label			Fields		Organization				
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse		
Pag1												
			form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Pag2												
			Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Blocks not shown												
			Block 3	Block 3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open

Application Block Display configuration

Block

- **Name:** The name that identifier the Block.
- **Label:** Title of the block to display in the application.

Title

- **Display:** It controls the display of the block title.

Label

- **Display:** It controls the display of the field labels of the block.
- **Position:** Options to display label :
 1. **Above:** Display the label above the field.
 2. **Beside:** Display the label beside the field.
 3. **Below:** Display the label below the field.

Fields


- **Columns:** Amount of columns side by side in the block.
- **Position:** The way to display the fields in the block:
 1. **Below:** Display the fields one below the other respecting the number of columns.
 2. **Beside:** Display the fields one beside the other respecting the number of columns.
 3. **Line:** Display the fields one beside the other with no tabulation.

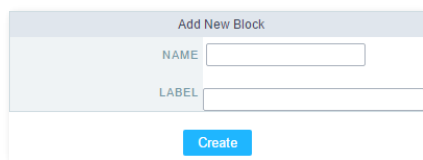
Organization

- **Next:** The way to display the blocks in the page:

1. **Below:** Set to show the following block below the current one.
 2. **Beside:** Set to show the following block beside the current one.
 3. **Tabs:** Set to show the following block in a different tab then the current one.
- **Width:** Set the block width in pixels or percentages. Use the symbol “%” to indicates the value in percentage.
 - **Collapse:** Enables the option to close the block.

Create A New Block

To include new blocks in an Application, click on the button . Then, enter the name and label of the block in the following interface and finish by click on Create.



The 'Add New Block' form contains two input fields: 'NAME' and 'LABEL'. Below these fields is a blue 'Create' button.

Creating application blocks configuration

Name

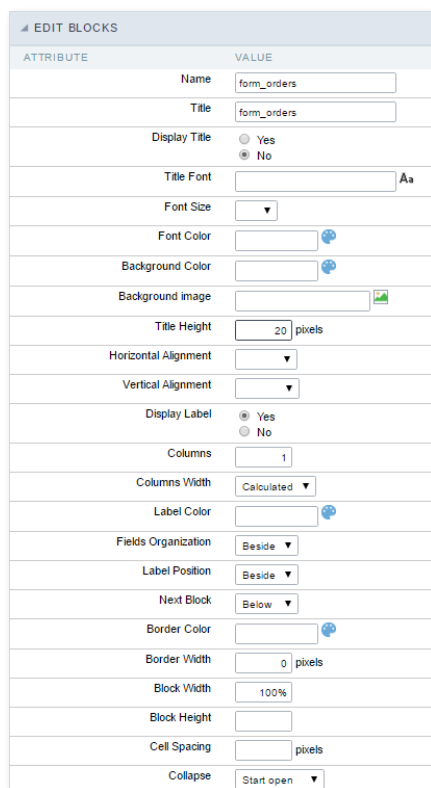
Name of the Block.

Label

Title of the block to display in the application.

Edit Blocks

To edit a block, click on the icon , that is on the left side of the block. Then you can see the following interface to define the parameters of the blocks. Click on Save to finish.



The 'EDIT BLOCKS' interface is a table with two columns: 'ATTRIBUTE' and 'VALUE'. It contains various settings for a block, including text, color, font, and layout options.

ATTRIBUTE	VALUE
Name	form_orders
Title	form_orders
Display Title	<input type="radio"/> Yes <input checked="" type="radio"/> No
Title Font	<input type="text"/> Aa
Font Size	<input type="text"/>
Font Color	<input type="text"/>
Background Color	<input type="text"/>
Background image	<input type="text"/>
Title Height	<input type="text"/> 20 pixels
Horizontal Alignment	<input type="text"/>
Vertical Alignment	<input type="text"/>
Display Label	<input checked="" type="radio"/> Yes <input type="radio"/> No
Columns	<input type="text"/> 1
Columns Width	Calculated
Label Color	<input type="text"/>
Fields Organization	Beside
Label Position	Beside
Next Block	Below
Border Color	<input type="text"/>
Border Width	<input type="text"/> 0 pixels
Block Width	<input type="text"/> 100%
Block Height	<input type="text"/>
Cell Spacing	<input type="text"/> pixels
Collapse	Start open

Application Block editing interface

Name

Name of the block. ##### Title
 Block title for display. ##### Display Title
 This option, when active, allows displaying the block title. ##### Title Font
 Set the font family of the block title. ##### Font Size
 Set the font size of the block title. ##### Font Color
 Set the font color of the block title. ##### Background Color
 Set the Background Color of the block title. ##### Background image
 Set a Background image for the block title. ##### Title Height
 Height in pixels of the block title line. ##### Horizontal Alignment
 Horizontal Alignment of the block title (Left, Center, and Right). ##### Vertical Alignment
 Vertical Alignment of the block title (Top, Middle, and Bottom). ##### Display Label
 Display the labels of the fields in the block. ##### Columns
 Amount of field columns in a block. ##### Columns Width
 Set the field column width of the block. ##### Label Color
 Color of the field labels. ##### Fields Organization
 The way to display the fields in the block. ##### Label Position
 Set the position of the field labels of the block. ##### Next Block
 Set the position of the following block relating to the current one. ##### Border Color
 The border Color for the block. ##### Border Width
 The border Width for the block. ##### Block Width
 The width for the block. ##### Block Height
 The Height for the block. ##### Cell Spacing
 The Cell Spacing in the block. ##### Collapse
 It enables the option to close the block.

Layout Settings

ScriptCase creates an app by using default values defined per project, but you can change those values for a specific application on this interface.

LAYOUT SETTINGS

Header Template	Flat	Template name used for the application header.
Footer Template	Default	Template name used for the application footer.
Button		Use different buttons than what was defined in the color scheme.
Themes	Sc9_Rhino	Use different themes from the one defined for the color scheme

Header

|< < > >| xxyyzz xxxxx yyyy

Block 1

Name: xxxxxxxxxxxx

Type: Male Female

Address*: xxxxxxxxxxxx

Groups*: Male Female

Countries: Afghanistan

Address: yyyyyyyyyyyyyyy

Photos:

Drag & Drop files here

Image1.png ✓
Image2.png ✗

Captcha:

Application Layout Settings

Header Template

Allows choosing the look of the application Header.

Footer Template

Allows choosing the look of the application Footer.

Button

Allows choosing the button theme for the application.

Themes

Choose one of the selected themes in the project properties. It defines the look of the application, like colors, fonts, and others.

Header & Footer

Header

Here we can define if we want to display the header or not, the application title, and fill the header variables. The header looks according to the selected template in the Layout settings interface.

HEADER	
ATTRIBUTE	VALUE
Display Header	<input checked="" type="checkbox"/>
Insertion Title	{lang_othr_frmi_tit} - orders
Update Title	{lang_othr_frmu_tit} - orders
Header Variables	
SC_HEADER	Title ▾
SC_VALUE	Date ▾ <input type="text"/> ?

Application Header Configuration

Display Header


This option determines if we want to display the header or not.

Title

Allows informing the title of the application.

Header Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **Field:** When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **Title:** It displays the value of "**Application Title**" in the header.
- **Date:** It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **Image:** It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **Value:** It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Depending on the Application, you may have more than one title option.

Footer

FOOTER	
ATTRIBUTE	VALUE
Display Footer	<input type="checkbox"/>
Footer Variables	
NM_FOOTER_DATA	▾
NM_FOOTER_LOGO	▾
NM_FOOTER_TITLE	▾

Application Footer Settings


This page may change depending on the footer template chosen in the Layout Settings.

Display Footer

This option determines if we want to display the footer or not.

Footer Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **Field:** When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **Date:** It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **Image:** It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **Value:** It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Dynamic Search

This feature allows the end-user to search for records without leaving the current screen. The developer must inform the available fields for search.

Settings

DYNAMIC SEARCH SETTINGS	
ATTRIBUTE	VALUE
Use the ENTER key to	<input type="text"/>

Dynamic Search Settings Interface

- **Use the ENTER key to** : It allows you to define the action that the Enter Key has on the Search form. **Tabulate** enables you to navigate between fields, and **Submit** performs the search(activates the Search button).

Select Fields

This interface allows to select the available fields in the Dynamic Search.

SEARCH DYNAMIC FIELDS		
ATTRIBUTE	VALUE	DESCRIPTION
Select the fields that will be used for dynamic search.		
<ul style="list-style-type: none"> customerid companyname contactname contacttitle birthdate country regionid stateid city address postalcode phone fax cityid creditlimit cardtype cardnumber notes 	<ul style="list-style-type: none"> ⊗ ⊕ ⊖ ⊗ 	<ul style="list-style-type: none"> grid_customers

Dynamic Search Field Selection.

Fields

This is a list of fields available for use in the filter. Click on the desired field to access the configuration instructions.

Text Field	Field Select
Integer Field	Field Double Select
Decimal Field	Field Checkbox
Currency Field	Field Radio
Date Field	Field Text Auto-Complete
Field time	Field Number Auto-Complete
Field Datetime	

Settings

Navigation

This interface allows defining the navigating behavior of the application

ATTRIBUTE	VALUE
Exit URL	<input type="text"/> 🔗
Close on Exit	<input type="checkbox"/>
Redirect URL	<input type="text"/> 🔗
Redirect Variable	<input type="text"/>

Navigation Interface.

Exit URL

URL to where the user goes when he clicks on the "exit" button.

Close on Exit

Close the browser window when the user clicks on the "exit" button.

Redirect URL

Redirect to another URL in case there aren't any global variables available.

Redirect Variable

Creates a variable with the application name and sends it to the redirected application.

Global Variables

This screen shows all global variables used in the application

The global variable is an external parameter required for running the application. The application can include global variables in the WHERE clause, field definitions and names, event programming, and so on.

You can define global variables by using square brackets ([variable]). You can pass the parameters to the application through one of the methods: Session, Post, and Get

if you need to use [Database Identifiers](#) in or tables, we recommend using the double quotation marks(" ") instead of the brackets([]) because of the conflict with the [Global Variables syntax](#) of Scriptcase. Using the brackets as database identifiers may lead to problems in the Scriptcase applications.

APPLICATION
<ul style="list-style-type: none"> ⚙ Settings 🔗 Navigation 📄 Messages 🌐 Global Variable

Global variables Interface.

Example:

```
Select CustomerID, CustomerName from Customers WHERE CustomerID = '[v_customerid]'
```

We do not recommend using session variables (global variables) with the same name as the table fields.

In this case, the variable **v_customerid** is displaying in the global variables configuration.

ATTRIBUTE	VALUE
global	<div style="border: 1px solid #ccc; padding: 5px;"> <p>Scope</p> <p><input type="checkbox"/> SESSION</p> <p><input checked="" type="checkbox"/> POST</p> <p><input checked="" type="checkbox"/> GET</p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 5px;"> <p>Settings</p> <p><input type="checkbox"/> Optional</p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 5px;"> <p>Type</p> <p><input type="radio"/> Out</p> <p><input checked="" type="radio"/> In</p> </div>

Global variables configuration Interface.

Attribute

It shows the variable's name in the application.

Value

Defines the behavior of the variables, divided into three blocks, they are:

Scope

Defines how the application receives the variable. A variable defined as the POST method in the app does not receive value if it comes with the GET method. Those are the methods available:

- **ession**: Defines that the variable must be created as a PHP session variable by another application.
- **ET**: Defines that the variable must be passed by the URL, that is, visible by the browser.
- **OST**: Defines that the variable must be passed through the POST method of an HTML form.

Settings

- **ptional**: Check it to don't validate the variable at runtime.

Type (In / Out)

Defines whether the variable comes from another application (In), or it is starting in the current app (Out).

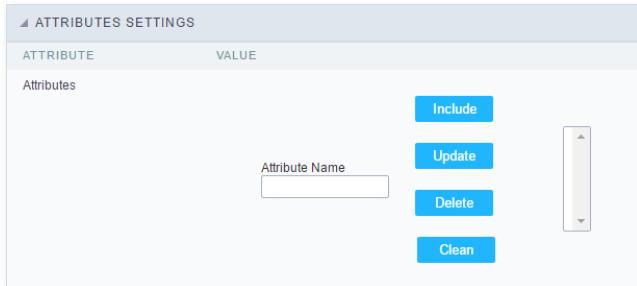
Description

Indicates where the application is using the variable.

In this version of ScriptCase is incorporated the concept of programming, with the use of attributes, methods, resources and libraries. In the previous version it was already possible to create business rules in applications using this concept the big difference now is that this can be done in a more organized, facilitating both the development as the understanding of the rule by another developer.

Attributes

The attributes are variables that has the global scope declared in the application. An attribute that can be referenced in all the methods and the application's events.



Attributes configuration Interface.

- **attributes**
 - **attribute Name** : This field allows to inform the attributes name, after typing the name just click on **include**. In case you want to **update** click on the attributes name (located on the buttons' right side), do the necessary modifications and click on update. In case it is necessary to **delete** an attribute, it is necessary to do same process as the update, but when selecting the attribute, it is necessary to click on the delete button. The **clear** button, as the name suggests, it clears the list of all the attributes listed on the right.

Internal Libraries

To use the internal libraries just select the correspondent (Project Libraries for example). This process allows the methods existing in the library, to be visible in all the application's Events and Methods. See how to manage the libraries by [clicking here](#).

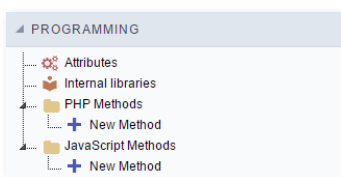


Internal Libraries management Interface

PHP Methods

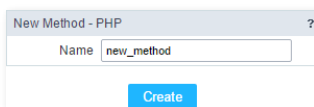
Methods are function or procedures declared by the developer, that helps when applying the routines. Using methods in the applications allows to reuse your code throughout the application, optimizing the development experience.

Creating A New Method



- Give the method a name and click Create. As in the image below.

PHP methods are incorporated into the same class, so it is not possible to use two functions with the same name even when used in different methods.



- Methods can receive parameters.


```
function new_method
1 echo "Hello World!!!";
```

- dd the amount of variables:

Definition of the parameters of the method:

new_method

- No defined parameter.

Add 1 Parameter(s) Cancel

Save

- defining the variables:

Insertion of Parameters

Name	Type	Value Standard
	For Value	
	For Value	
	For References	

Save Back Cancel

- **ame** : Type in the variable's name.
- **ype** : Selecting the type of variables: For Value or For Reference.
- **alue Standard** : The parameter's value used to initialize when calling the method.

References allows to create a second name for a variable that you can use to read and modify the variable's original information.

- diting a parameter:

Definition of the parameters of the method:

new_method

Parameters
Stest = test

☑ ☐ ✎ ✕

Add 1 Parameter(s) Cancel

Save

- ☑ : Checking all the parameters in the parameter list.
- ☐ : Uncheck all the parameters in the parameter list.
- ✎ : Edit the selected parameter of the list.
- ✕ : Deletes the selected variable of the list.

Preview

In this area you can customize the chart themes, you will be able to preview how it will look like according to each chart type.

If you customize the theme using this area, it will be available only for the current application only. If you need to customize the theme and apply to other application, project and users, then you will need to use the "Chart themes" option within the main menu "Layout > Chart themes". There you will find the same interface with the option to save the changes as "user level, project level or Scriptcase level"

Preview chart as 2D Column, single series

Choose theme

sc_Pastel Save

- Border
- Background
- Canvas
- Title
- X-axis
- Y-axis
- Data
- Palette
- Data title
- Data values
- Div Lines & Grids
- Anchors
- Tooltip
- Tick marks
- Legend

Themes preview

Border

Activate this function to add a border around the chart area.

Preview chart as 2D Column, single series

Choose theme

sc_Pastel Save

Border

Border use
Use

Color
#0b0a0a

Width
9

Opacity
51

Background

Canvas

Border.

- **order use** : Not informed, use or do not use.
 - **ot informed**: This option will use the default option of the theme.
 - **se** : Select the border as active, you will need to set the width.
 - **o not use** : to remove or do not use the border.
- **olor** : Select the border color, and may report a hexadecimal value or select using the color picker.
- **idth** : Sets the border width in pixels.
- **pacity** : Sets the border transparency.

Background

Options to configure the background color of the chart.

Preview chart as 2D Column, single series

Choose theme

sc_Pastel Save

Border

Background

Color

#00e34e Color picker

#161414 Color picker

New color

Opacity

Canvas

Title

Background

- **olor** : Select the background color, and may report a hexadecimal value or select using the color picker.
- **ew color** : Adds a new color for the background.
- **pacity** : Select background transparency.

Canvas

Options to configure the chart canvas.

Preview chart as 2D Column, single series

Border

Background

Canvas

Background color

#f5f5dc Color picker

New color

Background alpha

29

Border use

Do not use

Border Color

Color picker

Border width

Border alpha

Title

Canvas

- **ackground color** : Selects the color of the canvas, and may report a value hexadecimal or choose picker.
- **ew color** : Adds a new color option.
- **ackground alpha** : Selects the background transparency.
- **order use** : Not informed, use or do not use.
 - **ot informed**: This option will use the default option of the theme.
 - **se** : Selects the border as active, you will need to set the width.
 - **o not use** : To remove or to do not use the border.
- **olor** : Selects the canvas color, and may report a hexadecimal value or select using the color picker.
- **idth** : Sets the border width in pixels.
- **pacity** : Sets the canvas transparency.

Title

Options to configure the formatting of the title.

Preview chart as: 2D Column, single series

Monthly revenue for last year
Harry's SuperMart

Month	Revenue (in USD)
Jan	420K
Feb	810K
Mar	720K
Apr	550K
May	910K
Jun	510K
Jul	680K
Aug	620K
Sep	610K
Oct	490K
Nov	900K
Dec	730K

Title font
Tahoma, Geneva, sans-serif
B 24 [Color: Red]

Subtitle font
Arial, Helvetica, sans-serif
B 16 [Color: Green]

Alignment
Left Show on top

X-axis
Y-axis

Title

- **title font** : Selects the font, size, color, and formatting of the title.
- **ubtitle font** : Selects the font, size, color and format of the subtitle to left, center or right.
- **lignment** : Selects the position of the title and subtitle.
- **how on top** : If this option is selected the title and the subtitle will be positioned above the chart, otherwise will be positioned below.

Axes

This option configures the formatting of the texts of the x and y axis

Preview chart as: 2D Column, single series

Monthly revenue for last year
Harry's SuperMart

Month	Revenue (in USD)
Jan	420K
Feb	810K
Mar	720K
Apr	550K
May	910K
Jun	510K
Jul	680K
Aug	620K
Sep	610K
Oct	490K
Nov	900K
Dec	730K

X-axis

Font
"Courier New", Courier, monospace
B I 13 [Color: Black]

Font alpha
[Slider]

Background color
[Color Picker]

Background alpha
[Slider]

Border Color
#a1212 [Color Picker]

Border width
3 [Slider]

Border alpha
65 [Slider]

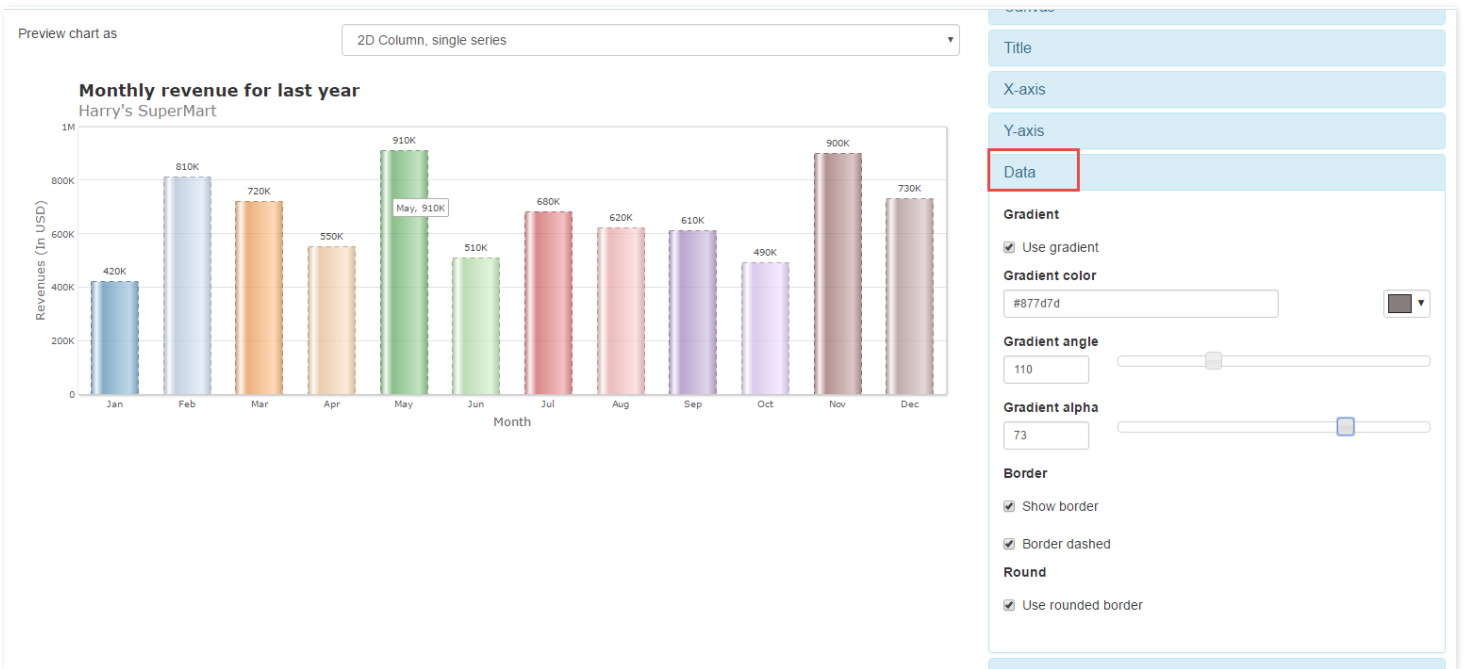
Border dashed

Axes

- **ont** : Selects the font, size, color, and formatting of the axis.
- **background color** : Selects the background color, and may report a hexadecimal value or select using the color picker.
- **ackground alpha** : Selects the transparency of the background color of the axis.
- **order Color** : Selects the color, and may report a hexadecimal value or select using the color picker.
- **order width** : Sets the width in pixels.
- **order alpha** : Sets the transparency.
- **order dashed** : If this option is selected, the border is dashed, this option is only valid if the option "Display" is checked.

Data

Options to set the chart's data (bank records).

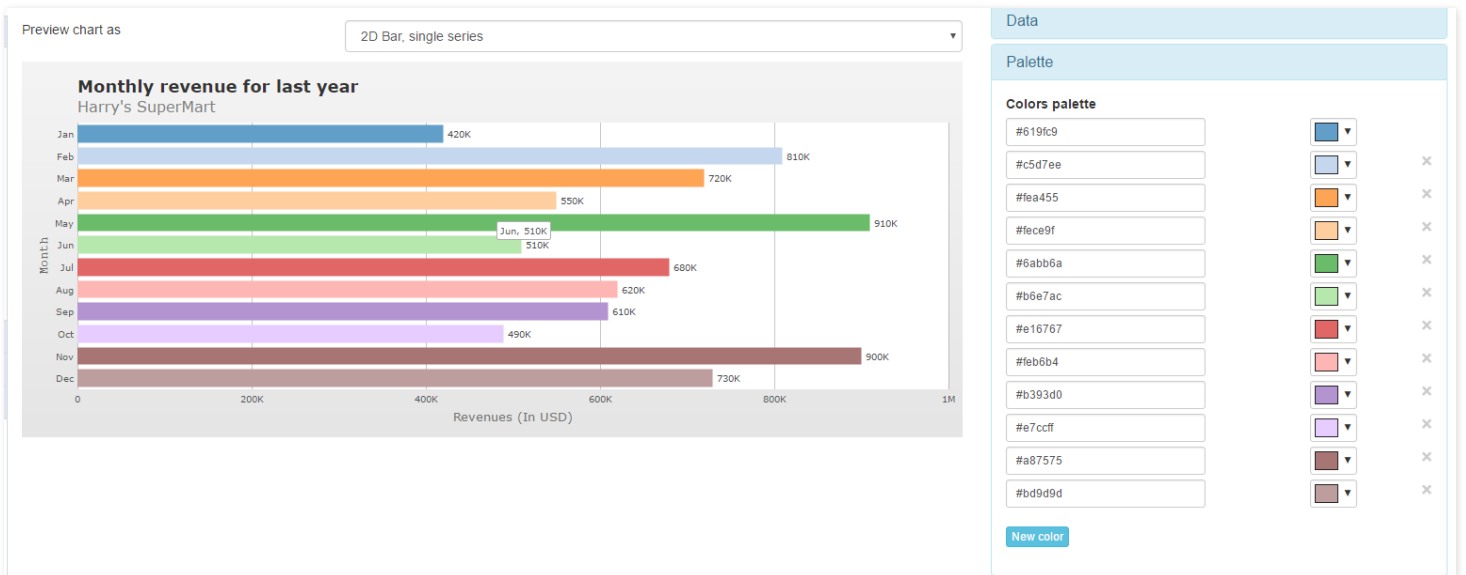


Data

- **se gradient** : Enables the data gradient.
- **radient color** : Selects the gradient color, and may add a hexadecimal value or select using the color picker.
- **radient angle** : Selects the angle of the gradient.
- **radient alpha** : Selects the gradient transparency.
- **how border** : If this option is selected is activated the border in the data.
- **order dashed** : If this option is selected, the border is dashed, this option is only valid if the option "Display" is checked.
- **se rounded border** : This option leaves the rounded border.

Palette

Options to choose the colors of each chart palette.

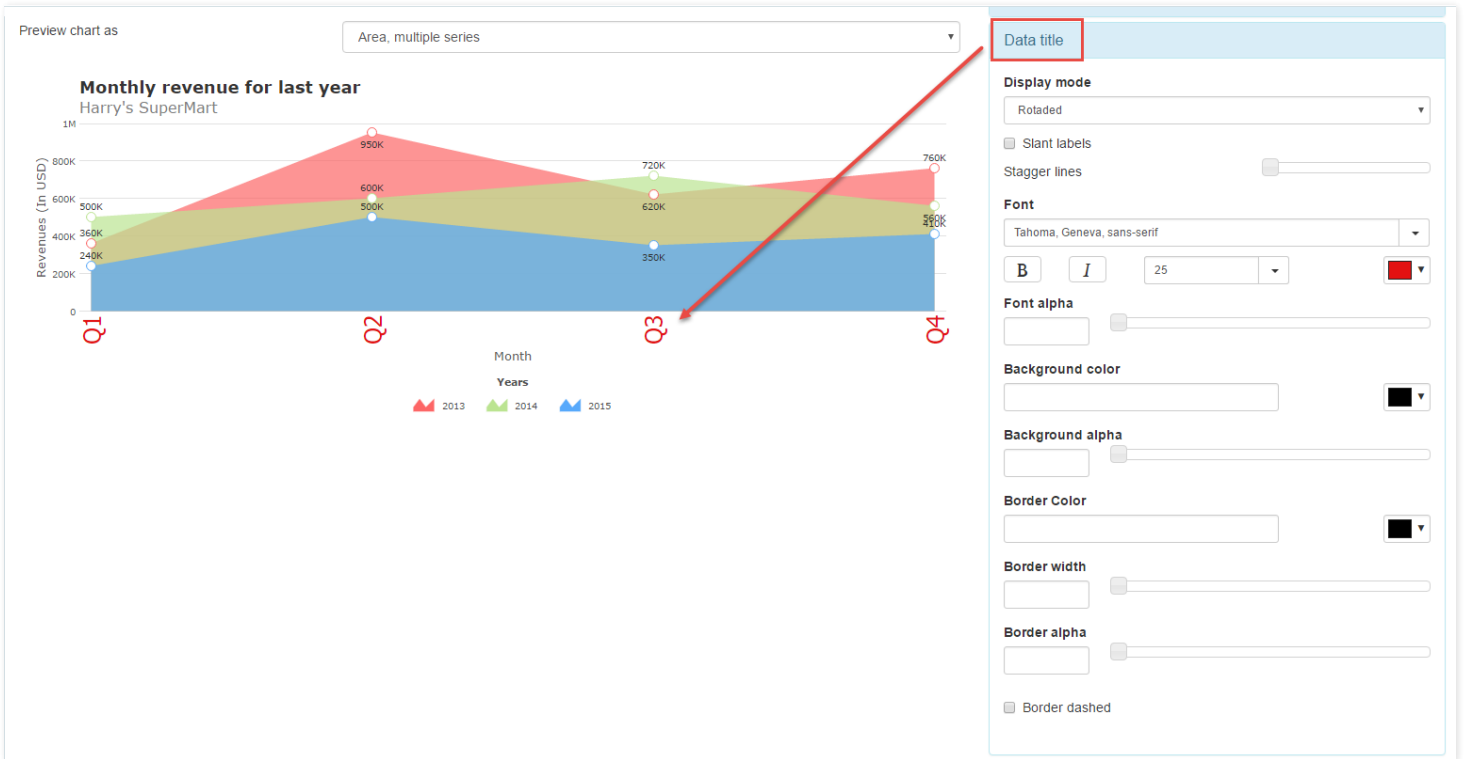


palette

- **olors palette** : Select the color of each chart palette.

Data title

Options to configure title formatting of data.

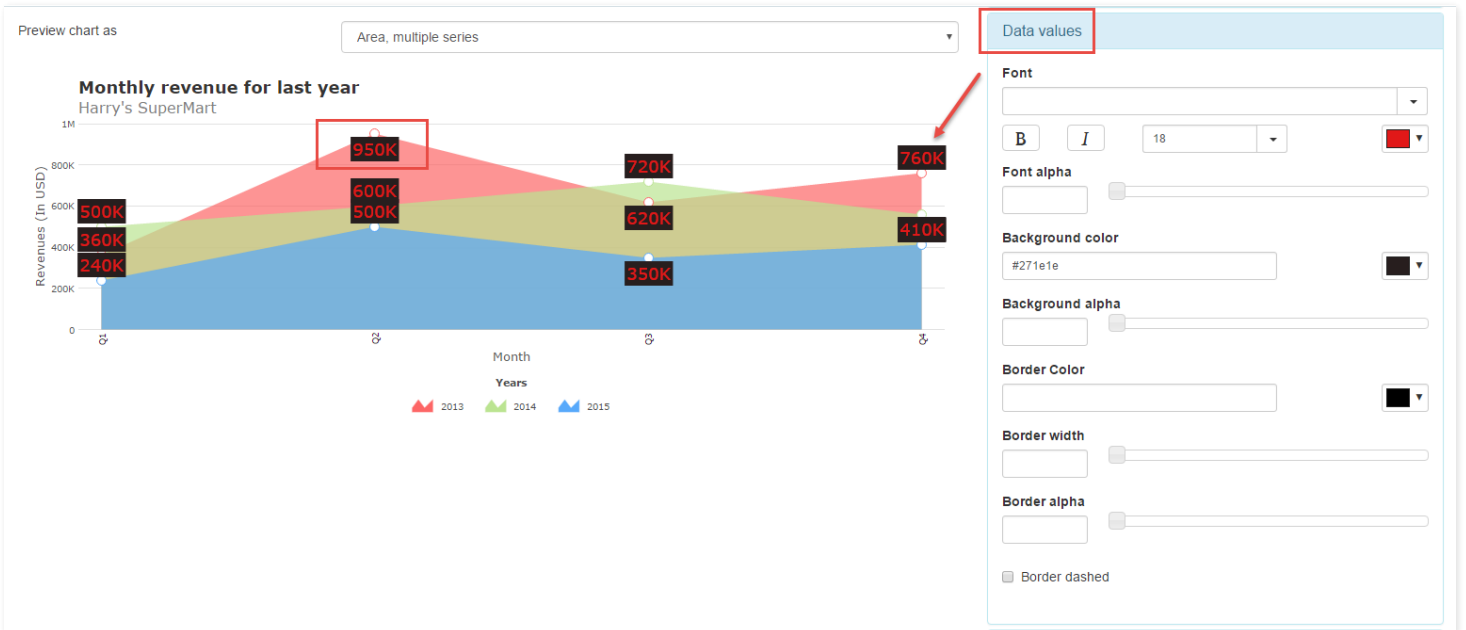


Data title

- **display mode** : Selects the display mode to "automatic", "rotated", "line break", "in levels" and "None", if the Automatic option is selected, select the default option.
- **slant labels** : Selects the number of levels to the data.
- **font** : Selects the font, and your formatting such as bold, italic, size and color of data.
- **font alpha** : Selects the transparency of data title font.
- **background color** : Select the background color, and may add a hexadecimal value or select using the color picker.
- **background alpha** : Selects the transparency of the background color of the title of data.
- **border width** : Sets the width in pixels.
- **border alpha** : Sets the transparency.
- **border dashed** : If this option is selected, the border is dashed, this option is only valid if the option "Display" is checked.

Data values

Options to configure the formatting of data values.



Data values

- **font** : Selects the font, and your formatting such as bold, italic, size and color of the data values.
- **font alpha** : Select the transparency of the source of the data values.
- **background color** : Select the background color, and may add a hexadecimal value or select using the color picker.
- **background alpha** : Selects the transparency of the background color of the title of data.

- **order Color** : Selects the border color for the data values.
- **order width** : Selects the border width for the data values.
- **order alpha** : Selects the transparency of the color of the border of the data values.
- **order dashed** : If this option is selected, the border is dashed, this option is only valid if the option "Display" is checked.

Div Lines and Grids

Options to configure the Division of horizontal and vertical lines on the graph.

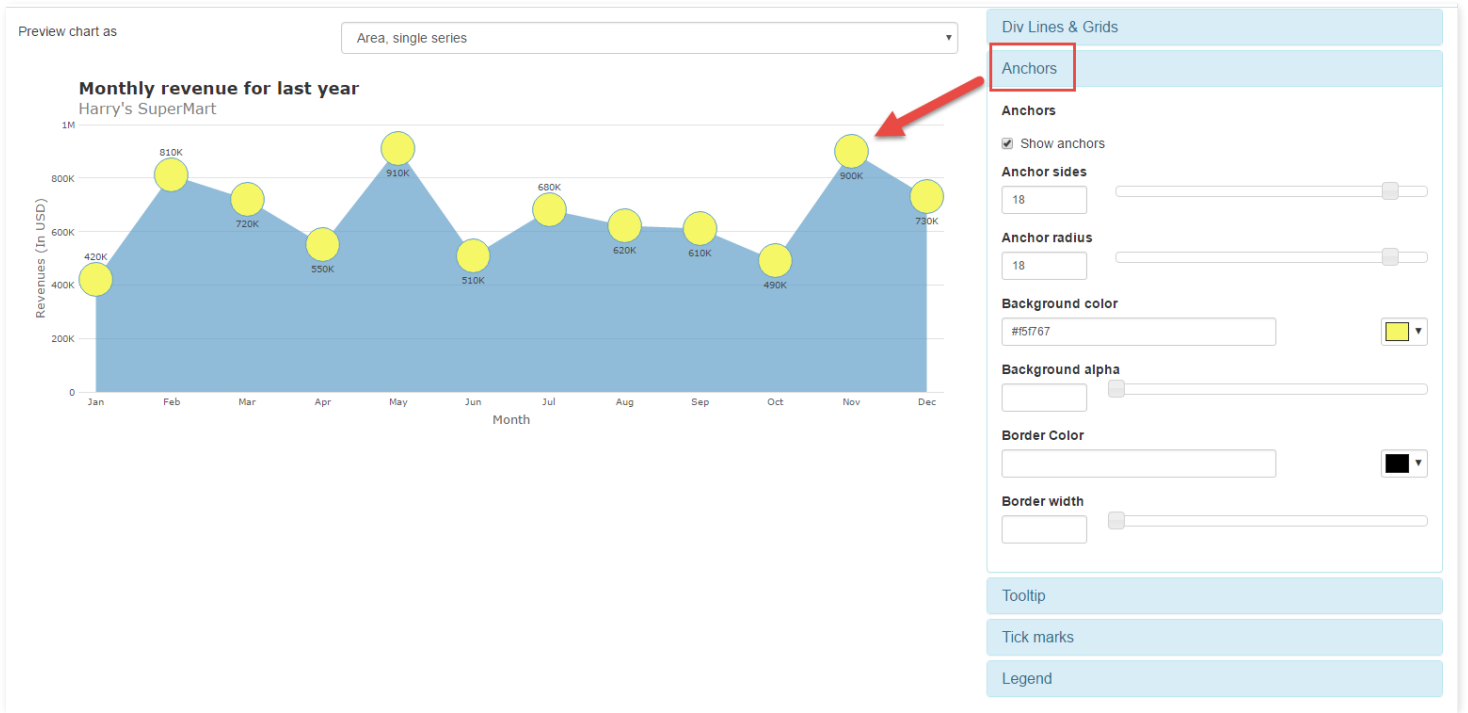


Div Lines and Grids

- **mount of horizontal lines** : Selects the amount of horizontal lines that will be shown in the chart.
- **horizontal line color** : Selects the horizontal line color.
- **horizontal line thickness** : Selects the thicker horizontal line.
- **horizontal line alpha** : Selects transparency horizontal line
- **horizontal line dashed** : If this option is selected will horizontal line plot.
- **olor horizontal stripes** : If this option is selected you can choose the horizontal line color.
- **horizontal stripes color** : Selects the color you wish to case "Coloring horizontal Ribbon" is checked.
- **horizontal stripes alpha** : Selects transparency horizontal ribbon.
- **mount of vertical lines** : Amount of vertical lines
- **vertical line color** : Selects the color of the vertical line.
- **vertical line thickness** : Selects the vertical line thicker.
- **vertical line alpha** : Select the transparency of the vertical track.
- **vertical line dashed** : If this option is selected will draw the vertical line.
- **olor vertical stripes** : If this option is selected the developer can choose the color of the vertical line.
- **vertical stripes color** : Selects the color you wish to case "Color vertical stripes" is checked.
- **vertical stripes alpha** : Select the transparency of the vertical stripes.

Anchors

Options to configure anchors on the chart.

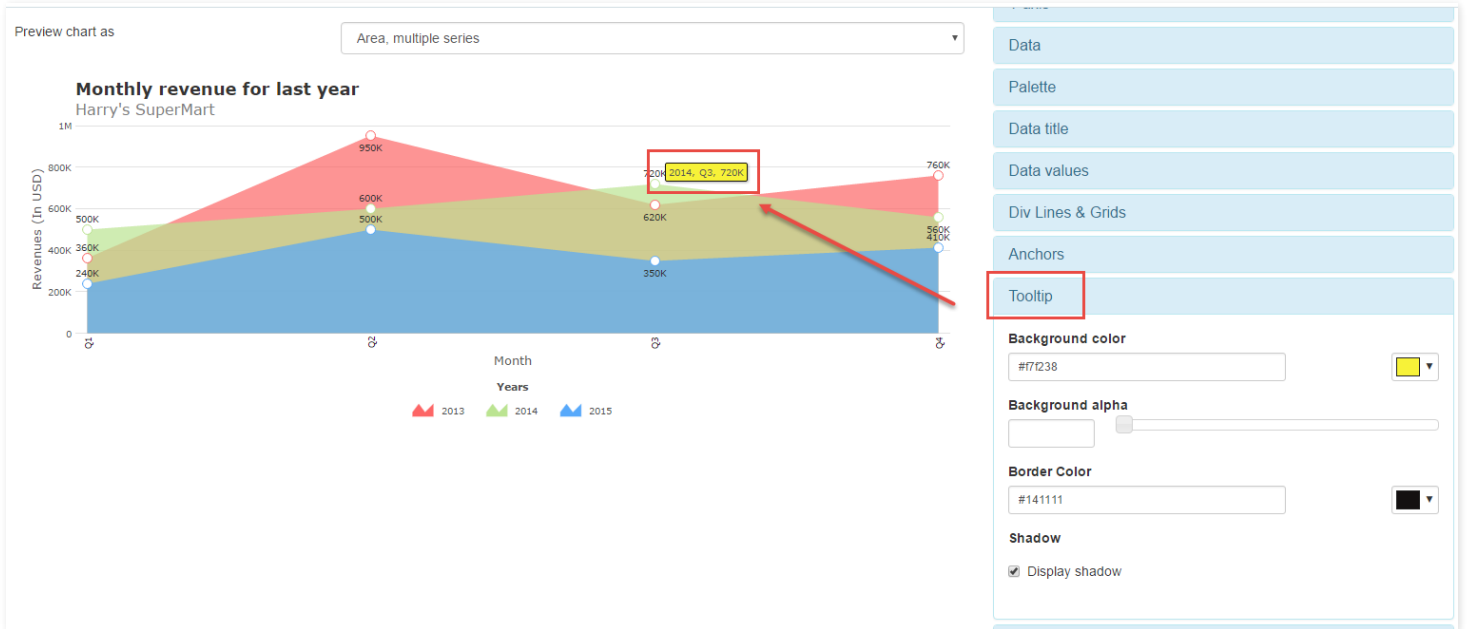


Anchors

- **show anchors** : Enables the anchors in the chart.
- **anchor sides** : Select the width of the anchor.
- **anchor radius** : Selects the size of anchor beam.
- **background color** : Select the color, and may add a hexadecimal value or select using the color picker.
- **background alpha** : Selects the transparency of the anchor.
- **border Color** : Select the anchors border color, and may add a hexadecimal value or select using the color picker.
- **border width** : Select the border width of the anchor.

Tooltip

Options to configure the tooltips in the chart.



Tooltip

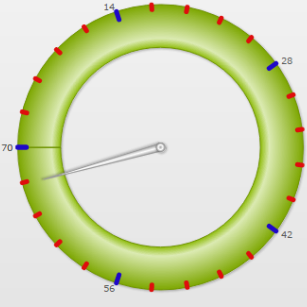
- **background color** : Select the tooltip color, and may add a hexadecimal value or select using the color picker.
- **background alpha** : Select the transparency of the background of the tooltip.
- **border Color** : Select the tooltip border color, and may add a hexadecimal value or select using the color picker.
- **shadow** : Enable the shadow in the tooltip.

Tick marks

Option to the configure chart marks, this option is only possible for the "Gauge" charts.

Preview chart as
Angular Gauge ▾

Monthly revenue for last year
Harry's SuperMart



Tick marks

Tick marks

Show tick marks

Place ticks inside

Tick mark values

Show tick values

Show limits

Place values inside

Major tick color

▾

Major tick alpha

Major tick thickness

Major tick height

Minor tick color

▾

Minor tick alpha

Minor tick thickness

Minor tick height

Tick marks

- **how tick marks** : Enables the tick marks on the chart.
- **lace ticks inside** : If this option is checked the marks of the chart will tile chart poker.
- **how tick values** : Show tick values
- **how limits** : If this option is selected will show the chart limits.
- **lace values inside** : If this option is selected will show the values in the chart
- **ajor tick color** : Major tick color.
- **ajor tick alpha** : Major tick transparency
- **ajor tick thickness** : Major tick thickness.
- **ajor tick height** : Select a height for the major tick.
- **inor tick color** : Select a color for the minor tick
- **inor tick alpha** : Select the transparency for the minor tick
- **inor tick thickness** : Minor tick thickness.
- **inor tick height** : Select a height for the minor tick.

Legend

Options to configure the chart legend.

Preview chart as 2D Column, multiple series

Monthly revenue for last year

Harry's SuperMart

Quarter	2013	2014	2015
Q1	360K	500K	240K
Q2	950K	600K	500K
Q3	620K	720K	350K
Q4	760K	560K	410K

Legend

Position
Below

Allow drag

Icon scale
2

Scroll background color
#000000

Title font
"Courier New", Courier, monospace
18

Item fonts
"Courier New", Courier, monospace
B 18

Background color
#FFFFFF

Background alpha
0

Border Color
#0e0d0d

Border width
0

Border alpha
0

Legend

- **osition** : Selects the position of the legend, "beside" or "Below"..
 - **eside** : Positions the legend beside the chart.
 - **elow** : Positions the legend below the chart.
- **llow drag** : If this option is selected it will be possible to drag the chart legend to the desired position.
- **roll background color** : Selects the color of the scroll bar.
- **itle font** : Select the font, and your formatting such as bold, italic, size and color of the legend.
- **ackground color** : Selects the background color of the legend.
- **ackground alpha** : Selects the transparency of the legend.
- **order Color** : Select the legend border color, and may add a hexadecimal value or select using the color picker..
- **order width** : Selects the Border width of the legend.
- **order alpha** : Selects the border transparency of the legend.

In the limit option, the user can choose the maximum limit that will be displayed in the chart.

Limit settings ?		
ATTRIBUTE	VALUE	DESCRIPTION
Field	<input type="text" value="id"/>	Field to perform limit filter
Summarization	<input type="text" value="Sum"/>	In which summarization function the limit will be applied
Limit type	<input type="text" value="Top"/>	Set if the limit will be realized for the first or for the last.
Quantity	<input type="text"/>	Number of records to be applied for the limit

Chart limit Interface

Limit Settings

- **ield** : Field that will realize the filter in the desired limit.
- **ummarization** : In which summarization function will be done in the limit can be "Sum", "Maximum", "Minimum", "Average", "Variance", "Standard deviation", "Count" and "Distinct count".
- **imit type** : Defines if the limit will be realized by the firsts or by the last records.
- **uantity** : Quantity of records that will be applied in the limit.

CUSTOMERS_NAME_1	CUSTOMERS_ID_1	CUSTOMERS_EMAIL_1
<input type="text" value="Caio Vinicius (1)"/>	<input type="text" value="1 - 6"/>	<input type="text" value="caio@netmake.com.br (1)"/>
<input type="button" value="Limpar"/>		

Generated application

It allows you to send the generated export file by email. For that, you may be using SMTP or integration with **Mandrill** and **SES**.

For more information about Mandrill [click here](#)

For more information about Amazon SES [click here](#)



Sending Options

We must configure how to send the email or select an API already set in Tools> API. [Click Here](#) and see how to configure.

SMTP

SMTP is the standard protocol for sending emails over the Internet, and each provider has its SMTP.

For more information about SMTP settings [click here](#)

Sending settings		
ATTRIBUTE	VALUE	DESCRIPTION
API	- Custom -  	API for sending e-mails.
Gateway	smtp	
SMTP server	smtp.example.com	SMTP server to send email.
SMTP port	465	SMTP server port (example: 25, 465, 587).
SMTP user	root	User to connect to the SMTP
SMTP password	*****	SMTP server connection password.
SMTP protocol		Encryption protocol used by the SMTP server.
From email	default@example.com	Default shipping email.
From name	default	Default shipping name.

- **PI:** Allows you to select an API that is already configured in **Tools> API** or set a new one here by selecting **- custom -**. In this case you will see some options according to the selected Gateway.

SMTP Server

SMTP server address for the used provider.

Port SMTP

Define the Port of the mail server. Use the port 465 for security with SSL, 587 for security with TLS, or port 25 for port without security. By default, Scriptcase uses port 25.

User SMTP

SMTP server user.

Enter SMTP

SMTP server user password.



Protocol SMTP

Select the security protocol. If no value is declared, Scriptcase uses the No Security protocol as default.

- **-mail:** Sets the origin email, which will send the emails.
- **ame:** Name that will be displayed in the sent email.

Mandrill

Mandrill is a transactional email API for MailChimp users, ideal for sending data-driven emails.

Sending settings		
ATTRIBUTE	VALUE	DESCRIPTION
API	- Custom -  	API for sending e-mails.
Gateway	mandrill	
API key	Your API	API connection key.
From email	default@example.com	Default shipping email.
From name	default	Default shipping name.

- **PI:** Allows you to select an API that is already configured in **Tools> API** or set a new one here by selecting **- custom -**. In this case you will see some options according to the selected Gateway.
- **PI KEY:** Enter the key you obtained when setting up your API.
- **-mail:** Sets the origin email, which will send the emails.
- **ame:** Name that will be displayed in the sent email.

Amazon SES

Amazon Simple Email Service ([Amazon SES](#)) is an email sending service designed to assist in sending marketing emails, notifications, and transactional messages.

Sending settings		
ATTRIBUTE	VALUE	DESCRIPTION
API	- Custom - <input type="button" value="🔄"/> <input type="button" value="✎"/>	API for sending e-mails.
Gateway	Amazon SES	
API key	Your Key API	API connection key.
API Secret	Your Secret API	API secret, obtained in association with API KEY.
Region	Region	API Connection Region.
From email	default@example.com	Default shipping email.
From name	default	Default shipping name.

- **PI**: Allows you to select an API that is already configured in **Tools> API** or set a new one here by selecting **- custom -**. In this case you will see some options according to the selected Gateway.
- **PI KEY**: Enter the key you obtained when setting up your API.

API Secret

Enter the secret access key of your Amazon SES account.

Region

Amazon SES has endpoints in several regions, to reduce network latency, inform the region of the endpoint closest to your application. [See the regions.](#)

- **-mail**: Sets the origin email, which will send the emails.
- **ame**: Name that will be displayed in the sent email.

Export Settings

In these settings, we define the settings of the email sending interface when exporting, and we can define a default email subject and body.

Export settings		
ATTRIBUTE	VALUE	DESCRIPTION
To	<input type="text" value="[mail]"/>	Enter the default value for the field 'To'.
It has copy	Yes	Sets whether to display the field 'copy' within exporting options.
Copy (cc)	<input type="text" value="[mailcopy]"/>	Enter a default value for the field 'Copy'.
It has blind carbon copy (bcc)	Yes	Sets whether to display the blind carbon copy field in export option.
Blind carbon copy	<input type="text" value="[mailbcc]"/>	Enter the default value for the field 'blind carbon copy'.
Title	<input type="text" value="{lang_export_email_subject} sales"/>	Default email subject text. You can change the lang variable or add fixed text.
Search of sales		
Body	<input type="text" value="{lang_export_email_body}"/>	Default text for the body of the email. You can change the lang variable or add a fixed text.
Hello, Please find attached with this email the report in %. Thank you and Have a Nice Day		

To

Enter the default destination email.

Use Copy (CC)

Defines whether or not to display the 'copy' field within the export options.

- **o**: There will be no possibility to send email with copies.
- **es**: The field will be displayed within the export options and we can define a default email for copying. The end user can view this email.
- **idden**: The field will not be displayed within export options and we can set a default email for copying.

Copy (CC)

Enter the default email for the copy field. This option can also be empty, or you can use a global variable containing an email list.

Using Hidden Copy (BCC)

Sets whether or not to display the 'hidden copy' field within the export options.

- **o**: There will be no possibility to send email with copies.
- **es**: The field will be displayed within the export options and we can define a default email for copying. The end user can view this email.
- **idden**: The field will not be displayed within export options and we can set a default email for copying.

Copy (BCC)

Enter the default email for the hidden copy field. This option can also be empty, or you can use a global variable containing an email list.

Subject

Default email subject text. You can change the lang variable or add static text.

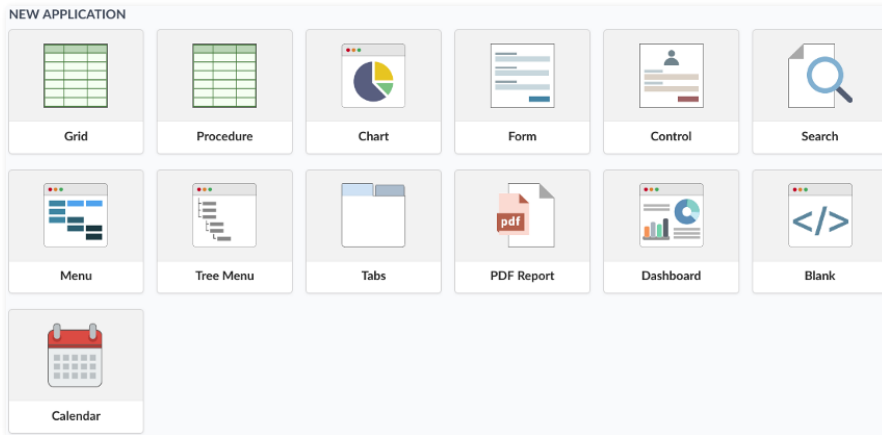
Color

Default body text for the email. You can change the lang variable or add static text.

New Application

The creation process of one application is simple enough and intuitive.

You just need to select which application you want to generate in the applications available in Scriptcase, That it will be generated automatically.



Application Data

When the application is selected, the following screen will be displayed below.

In this table you need to select the connection that will be used in the application. The tables will be loaded and displayed in the field **Table** when the connection is selected.

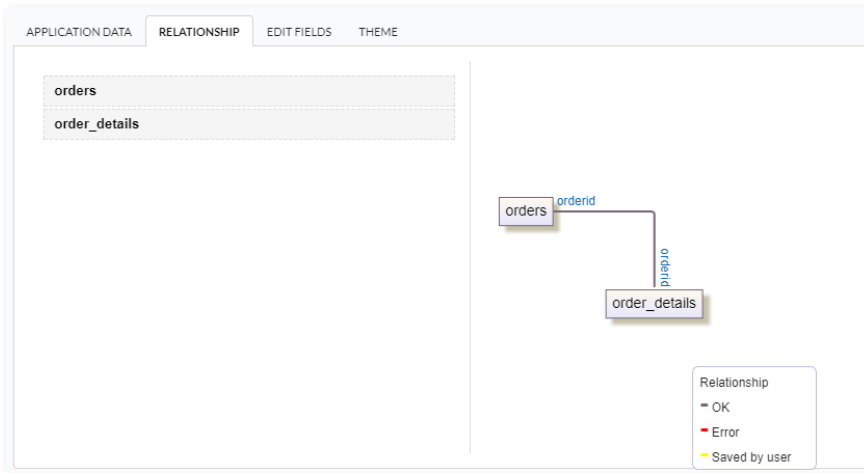
It is possible to select more than one table in the application creation.

After selecting the table and fields that should be part of the application, the name of the application will be formed by the type of application + name of the table. **Ex: grid_orders** However it is possible to change the application's name to any other desired name.

- **connection** - It defines the used connection to create the application. All the project's application will be displayed.
- **ables** - It defines the used tables in the application. (Form and Calendar can only use one table).
- **ields** - It defines the fields that will be part of the applications.
- **QL Select Statement** - Display the select statement created after selecting the tables and fields. This field allows also to insert a previously created SQL, if the used tables are available in the database selected in the connection.
- **ame** - Application's name that is being created, It can not contain special characters.
- **ocalization** - Application's Language. The project's default language will be automatically selected.

Relationship

When select two or more tables, the tab "Relationship" will be displayed. In this tab we can see the relationship created between the tables, where we can edit the related fields.



When we click in a link, in the screen above, it will displayed the related field's edition form, as you can see in the image below.

Edit table relation

ORDERS ORDER_DETAILS

orderid orderid +

Save Cancel

Edit Fields

In this tab will be displayed the fields that are going to be created in the application, also its labels and datatypes. We can make define some settings here like changing its labels and datatypes.

Fields	Label	Datatype
orderid	Orderid	Integer
customerid	Customerid	Text
employeeid	Employeeid	Integer
orderdate	Orderdate	Date
requireddate	Requireddate	Date

- **ields** - Name of fields stored in the database.
- **abel** - Name of fields displayed in the generated application.
- **atatype** - Field's Datatype.

Theme

In this screen you select the theme that the application will use. The default project's theme, that can be checked and changed in **Project > Properties**, will be selected automatically.

APPLICATION DATA | RELATIONSHIP | EDIT FIELDS | **THEME**

Sc9_Rhino

Header

◀ ◀ ▶ ▶ Add Save

Block 1.1

Title 1 Object text

Block 2.1

Title 1	Title 2	Title 3
11111111	22222222	33333333
11111111	22222222	33333333

Footer

- **heme** - List of available themes in the project.

After all the settings are done, click in the button **Create**. In the next screen the settings of the created application will be displayed.

Orientation models for the Form application, it can be: Single record, Multiple records, Editable grid and Editable grid view.

Single record

This form orientation allows you to edit only one table record at a time. You will be able to page the form to navigate record by record.

INITIAL SETTINGS

Orientation

Choose the Form orientation:

- Single record
- Multiple Records
- Editable grid
- Editable grid view

Form Orientation

Insert | **Cancel**

Name Email

Active Yes No Data dd/mm/yyyy

Maps Types Sport Auto

This form orientation allows you to edit only one table record at a time. You will be able to page the form to navigate record by record.

Multiple records

Form orientation with various editable records per page. Update or delete processes will run on selected records via check box. You can set as many records per page as needed to update, to delete or to add new records.

Initial Settings

Orientation

Choose the Form orientation:

- Single record
- Multiple Records
- Editable grid
- Editable grid view

Form Orientation

Save Selecteds | **Delete Selecteds**

Name	Email	Active	Data	Maps	Types
<input checked="" type="checkbox"/>	<input type="text"/>	<input type="radio"/> Yes <input type="radio"/> No	<input type="text"/>	Google Maps	<input checked="" type="checkbox"/> Sport <input type="checkbox"/> Auto
<input type="checkbox"/>	<input type="text"/>	<input type="radio"/> Yes <input type="radio"/> No	<input type="text"/>	Google Maps	<input type="checkbox"/> Sport <input type="checkbox"/> Auto
<input checked="" type="checkbox"/>	<input type="text"/>	<input type="radio"/> Yes <input type="radio"/> No	<input type="text"/>	Google Maps	<input type="checkbox"/> Sport <input type="checkbox"/> Auto

Form orientation with various editable records per page. Update or delete processes will run on selected records via checkbox. You can set as many records per page as needed to update, to delete or to add new records.

Pagination Pagination method for the grid records.

Lines Per Page Number of record lines per page.

Records for insert Record quantity to be available when inserting.

Fixed label This option will freeze the column labels on top of the screen during the page scrolling.

- **agination** : Paging type for the multiple records form, you can choose if it is going to be partial or total.
- **ines Per Page** : Quantity of rows of records per page, this option is available only when the pagination is set as "Partial".
- **ecords for insert** : You can set here the quantity of rows of records per page in the insert mode in order to add multiple records at a time.
- **ixed Label**: Establishes the column label in the screen top during the page scrolling.

Editable grid

Form orientation with various editable records per page. Each record has the update and delete controls available in its own line.

Initial Settings

Orientation
Choose the Form orientation:

Single record

Multiple Records

Editable grid

Editable grid view

New

Name	Email	Active	Data	Maps	Types
Abc	None	<input type="radio"/> Yes <input type="radio"/> No	10/11/2010	Google Maps	<input checked="" type="checkbox"/> Sport <input type="checkbox"/> Auto
↑ ×		<input type="radio"/> Yes <input type="radio"/> No		Google Maps	<input type="checkbox"/> Sport <input type="checkbox"/> Auto
↑ ×		<input type="radio"/> Yes <input type="radio"/> No		Google Maps	<input type="checkbox"/> Sport <input type="checkbox"/> Auto

Form orientation with various editable records per page. Each record has its own options for update and delete.

Pagination Pagination method for the grid records.

Lines Per Page Number of record lines per page.

Fixed label This option will freeze the column labels on top of the screen during the page scrolling.

- **agination** : Paging type for the multiple records form, you can choose if it is going to be partial or total.
- **ines Per Page** : Quantity of rows of records per page, this option is available only when the pagination is set as "Partial".
- **ixed Label**: Establishes the column label in the screen top during the page scrolling.

Editable grid view

Form orientation with various editable records per page. All records come with its data displayed in a read-only format. These data are only available for update by selecting the Edit option in the record itself.

Initial Settings

Orientation
Choose the Form orientation:

Single record

Multiple Records

Editable grid

Editable grid view

New

Name	Email	Active	Data	Maps	Types
Abc	None	Yes	10/11/2010	Google Maps	Sport
↑ ×		<input type="radio"/> Yes <input type="radio"/> No		Google Maps	<input type="checkbox"/> Sport <input type="checkbox"/> Auto
↑ ×		<input type="radio"/> Yes <input type="radio"/> No		Google Maps	<input type="checkbox"/> Sport <input type="checkbox"/> Auto

Form orientation with various editable records per page. All records come with its data displayed as read-only. This data will be only available for update when the edit options placed in the record line itself is selected.

Pagination Pagination method for the grid records.

Lines Per Page Number of record lines per page.

Use modal form to edit Use modal form to edit lines

Fixed label This option will freeze the column labels on top of the screen during the page scrolling.

- **agination** : Paging type for the multiple records form, you can choose if it is going to be partial or total.
- **ines Per Page** : Quantity of rows of records per page, this option is available only when the pagination is set as "Partial".
- **se modal form to edit** : Use a modal form to edit the records when you click to edit the record.
- **ixed Label**: Establishes the column label in the screen top during the page scrolling.

RELATED VIDEOS

[Types of Forms](#)

[Ajax Processing](#)

[Default Values](#)

[Auto Complete](#)

This interface is useful for editing the main display settings of the app.

SETTINGS	
ATTRIBUTE	VALUE
Friendly URL	<input type="text"/>
Line break in title	<input type="checkbox"/>
Horizontal Alignment	Center ▾
Margins	<input type="text"/> Up <input type="text"/> Down <input type="text"/> Right <input type="text"/> Left
Table Width	<input type="text" value="100"/>
Table Width Unit	Percent ▾
Table Columns	Automatic ▾

Form Settings Interface.

Friendly URL

Allows to define a URL for the application different from the application name. You can use the following chars (a-z, A-Z, 0-9, -). You still can change it on the initial screen of the home project, through the "Friendly URL" column in the apps list.

Line break in title

Use it if you want to break the line on the field titles.

Horizontal Alignment

Allows you to set the alignment of the application on the page.

Margins

Allows to define the margins of the application (Right, Left, Up and Down) in pixels.

Table Width

The width of the form table. Scriptcase uses Plain HTML to generate applications by using tables lines and cells.

Table Width Unit

Measurement unit for the table width defined in the previous option, being: percentage, pixel, or automatic.

Table Columns

This parameter defines the column (fields) width of the table (application).

Labels width

When the previous option is set to "Provided" you must inform the width of the labels here.

Layout and Behavior

This interface allows setting the behavior of the app.

LAYOUT AND BEHAVIOR	
ATTRIBUTE	VALUE
Automatic tab	<input type="checkbox"/>
Highlight Text on Focus	<input checked="" type="checkbox"/>
Use Enter to	▾
Field with Initial Focus	▾
Highlight Field with Error	<input type="checkbox"/>
Use a template from the HTML Editor	<input type="checkbox"/>

Layout and Behavior configuration Interface.

Automatic tab

Changes the focus to the next field when the amount of characters reaches its defined limit.

Highlight Text on Focus

Highlights the field when selected.

Use Enter to

Allows to use the "Enter" key to pass the focus to the next field.

Field with Initial Focus

Determines the field starts with focus when accessing the application. This option doesn't work with fields that contain a watermark.

Highlight Field with Error

Focus the field with the error when submitting the form.

Use a template from the HTML Editor

Allows to use the TinyMCE editor. You can edit and create your HTML Templates.

This interface is useful for editing the field settings and their position to display.

Edit Fields

1	Fields	Label	Datatype	New	Update	Read-only	Required	PK	DB value (Insert)	DB value (Update)
5	PAGE: PAG1									
6	BLOCK: FORM_CUSTOMERS									
	customerid	Customerid	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
	companyname	Companyname	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	contactname	Contactname	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	contacttitle	Contacttitle	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	birthdate	Birthdate	Date	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	country	Country	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
7	PAGE: FIELDS NOT SHOWN									
	regionid	Regionid	Integer	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	stateid	Stateid	Select	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	city	City	Text	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	address	Address	Text	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

Edit Fields Configuration.

Fields

It allows accessing the field settings (a pencil icon on the left). You can change the field position by dragging them to the desired position. Drag a field to "fields not displayed" if you don't want it in the app.

Label

It defines the title of the field in the app. For example: if the field name in the database is fld_txt_customer_name, you can display the label "Customer Name".

Data type

It informs the data type of the field.

New

It defines if the field is available when inserting new records.

Update

It defines if the field is available when updating records.

Read-Only

It defines the field as a label. The user can't change its value.

Required

It defines if the field must contain a value.

PK

It defines the Primary Keys fields.

DB value (Insert)

Defines a default value for the field when inserting a new record, like an auto-increment, Date, DateTime, or IP.

DB value (Update)

Defines a default value for the field when updating a record, like an auto-increment, Date, DateTime, or IP.

Page

It shows the pages available in the application. All apps have a page, at least. Each page contains one or more blocks.

Blocks

It shows the blocks available in the application. Blocks contain fields. All apps have a block, at least. A block is displayed if it contains one or more fields.

Page Fields Not Shown

Here we can see the fields that are not in the application.

Observe that you can drag any line to the desired position, blocks, and page. Pages contain Blocks, and Blocks contain fields.

Display Settings

Settings of messages display.

DISPLAY	
ATTRIBUTE	VALUE
Markers positioning	Right
Display message	<input checked="" type="checkbox"/>

Display Interface.

Attributes ### Markers positioning Set the position of the markers that indicates required fields.

Display message
Set it if you want to display the message of the required field.

This interface allows to sort the fields positioning.

Select the fields that will be displayed.

orderid customerid employeeid orderdate requireddate shippeddate shipvia freight priceorder shipcountry shipregion shipstate shipcity shipname shipaddress shippostalcode	➤ ➤ ➤ ➤	Pag1 form_orders orderid customerid employeeid orderdate requireddate	⤴ ⤴
--	------------------	---	--------

Save Restore

Field Positioning

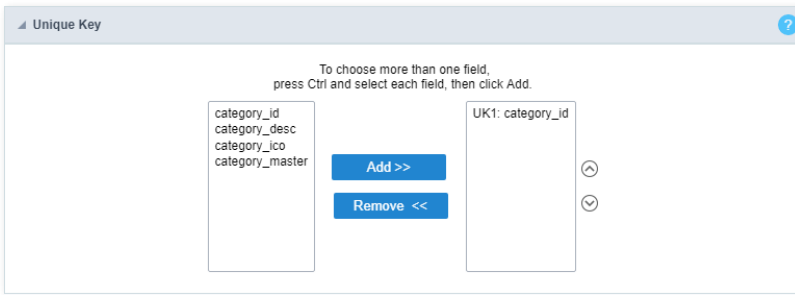
The left box has a list of all fields available in the application. Here you must select which fields you want to display in the app.

The right box contains the selected fields displayed in the application.

Use the arrows buttons between the boxes to select or deselect a field.

With this interface, you can inform which fields shouldn't have repetitions in the database. The app will return a message if the user tries to insert a repeated value into the field.

If you want to combine two or more fields as unique, select the fields by holding the CTRL key of your keyboard and then click on the "Add" button.



The screenshot shows a window titled "Unique Key" with a help icon. Inside, there is a list of fields on the left: "category_id", "category_desc", "category_ico", and "category_master". In the center, there are two buttons: "Add >>" and "Remove <<". On the right, there is a list box containing "UK1: category_id" and two arrow buttons (up and down) for scrolling.

Unique key configuration Interface.

The application toolbar has two segments: Top and Bottom, in a way that is possible to define to display buttons into both areas. Those areas work independently, allowing them to display the same button, for example.

It's also possible to select the buttons and their position if the application is running on a mobile device.

Toolbar

Desktop

Here we must inform the toolbar settings for the "Classic Web Version" mode and which buttons are available in the application when accessed from a **Desktop** environment.

Mobile

Here we must inform the toolbar settings for the "Mobile Version" mode. That is which buttons are available in the application when accessed from a **Mobile** dispositive.

Toolbar

Configure the toolbar below for a "classic web version " and also for a "mobile version".

Desktop

Mobile

Top Toolbar

Others
 QuickSearch
 Dynamic Search
 Languages
 Themes
 HelpCase
 Rows Counter
 Jump to
 Copy
Navigation
 Navigation by page
 First
 Previous
 Next

Left
 QuickSearch
 Dynamic Search
 Center
 Insert
 Cancel
 Update
 Delete
 Right
 Exit

Group

Add
Edit
Delete

Bottom Toolbar

Others
 QuickSearch
 Dynamic Search
 Languages
 Themes
 HelpCase
 Rows Counter
 Jump to
 Copy
Navigation
 Navigation by page
 First
 Previous
 Next

Left
 Jump to
 Center
 First
 Previous
 Navigation by page
 Next
 Last
 Right
 Rows Counter

Group

Add
Edit
Delete

Navigation:

Buttons relative to the navigation of the application.

Next	Move to the next page that can be a single record or a list of records.
Previous	Returns to displays the previous page records or a single record.
First	Move to the First page or record
Last	Move to the Last page or record
Exit	Close the application
Navigation by page	Displays a "page-number" navigation bar. Example: 1 2 3 4 5

Export:

The options available to export the Records. Scriptcase generates the following export formats for Forms:

PDF	Generates all the data of the application in a PDF format.
Print	Creates an HTML with the records ready for printing.

Update:

The CRUD options available in the Form.

Insert	Inserts the record into the database.
Update	Saves the changes made in a record.
Delete	Deletes the selected record.
Cancel	Cancel the changes made in a record before the insertion.

Others:

Other options available in the Form application.

Jump To	Move to the informed page or record.
Copy	Copy the current record data to another one.
Quick Search	Perform a quick search in the records of the application.
Dynamic Search	It displays the fields of the search to filter the records.
Languages	Displays a Combobox with the languages available in the project properties.
Themes	Displays a Combobox with the languages available in the project properties.
Rows Counter	Displays the number of records retrieved in the application.
HelpCase	Displays a button to open the help page.

Separator:

-----	Displays a line separating the buttons.
-------	---

Toolbar Mobile

The screenshot shows the 'Toolbar Mobile' configuration interface. It features two tabs: 'Desktop' and 'Mobile'. The 'Mobile' tab is selected, showing two sections for toolbar configuration: 'Top Mobile toolbar' and 'Bottom Mobile toolbar'. Each section contains a list of navigation and export options on the left and a list of left-side toolbar items on the right. A 'Group' dialog box is visible, allowing users to add, edit, or delete items from the toolbar.

Contains the same options as the **Desktop** version, adding only the item "Copy from Desktop", on which, when clicked, it copies the items from the **Desktop** toolbar to the **Mobile** toolbar.

Buttons Group

The **Group** option allows you to group a set of buttons of the application toolbar to display them as a dropdown, for example.

The screenshot shows the 'Buttons Group' configuration interface. It features the 'Toolbar Mobile' configuration with the 'Mobile' tab selected. A 'Copy from Desktop' button is visible. The 'Group of buttons' dialog box is highlighted with a red box, showing options to add, edit, or delete items from the toolbar.

Add

Add a new group of buttons.

Edit

Edit an existent group of buttons.

Delete

Delete the selected group of buttons.

When you press the **Add** or **Edit** option, you can see the settings to configure the grouper:

Edit

DISPLAY AS

DROPDOWN LIST THEME NAME

LABEL HINT \ TITLE

IMAGE BUTTON TYPE

DISPLAY DISPLAY POSITION

Display As

Allows displaying the group button as **Dropdown** or **Side by Side**.

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

Customer ID	Company Name	Phone	Credit Limit
dos		30074321	\$3,367.41
arilla		55554729	\$7,371.95
		55553932	\$6,757.53

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

Export

- PDF
- WORD
- Excel
- XML
- CSV
- RTF
- Print

Dropdown List Theme

Allows defining the Dropdown theme selecting between **Application theme** and **Button theme**.

Name

Allows defining a name for the button group.

Label

It is the displayed name for the button group in the application.

Hint\Title

Displays a hint to the end-user when the mouse is on the group of buttons.

Button Type

Allows displaying the button group as a Button, Image, or Link.

Image

Allows selecting an image for the button.

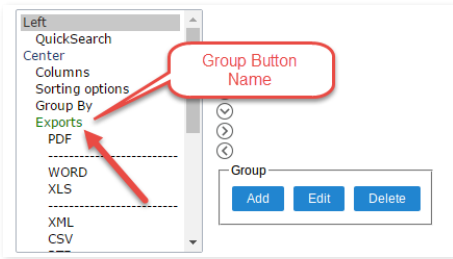
Display

Defines if the button displays only Text, only image, or both.

Display Position

Defines the position of the Text or Image (Text to the right, Image to the right).

After creating a button group, you need to move the grouped buttons below of the Button Group and then move them to the right. Like the image below:



Buttons Settings

Button	Label	Hint
QuickSearch		
Dynamic Search		
Insert		
Cancel		
Update		
Delete		
Exit		
Jump to		
First		
Previous		
Navigation by page		
Next		
Last		
Rows Counter		
New		

Button:

It displays the buttons available in the application.

Label:

Allows defining the labels of the buttons to display for the users.

Hint:

Allows defining the buttons hint that to display for the users.

Application Hotkeys

Scriptcase allows creating shortcut keys to your applications. You can select a predefined template or create specific actions for an application.

VALUE	DESCRIPTION				
<input checked="" type="checkbox"/> toggleHotkeys();	Define if the application will use hotkeys				
SC_DefaultHotkeys	Select the hotkey template from previously created schemas				
Clear <input type="button" value="+"/>					
<table border="1"> <thead> <tr> <th>ACTION</th> <th>KEYBINDING</th> </tr> </thead> <tbody> <tr> <td colspan="2" style="text-align: center;">No hotkeys configured</td> </tr> </tbody> </table>		ACTION	KEYBINDING	No hotkeys configured	
ACTION	KEYBINDING				
No hotkeys configured					
Clear <input type="button" value="+"/>					

Use hotkeys

Defines if the application uses hotkeys. When you enable this option, the default shortcut keys settings are disabled.

Hotkeys templat

Select the [hotkey template](#) previously created.

Action

Selects the triggered action when pressing the selected key.

Keybinding

Selects the keys responsible for executing the chosen action.

Add "+"

Adds a new action on the keys list.

Clear

It clears the selected hotkeys preference.

Options

Options	
ATTRIBUTE	VALUE
Format Row Counter	<input type="text"/>
The number of links displayed	<input type="text" value="5"/>
Help by Block	<input type="checkbox"/>
General Help	<input type="checkbox"/>
Toolbar buttons	<input type="text" value="A DIV below the toolbar. ▾"/>

Format Row Counter:

Allows defining the format of the row counter displayed on the application.

Example: (1 to 10 of 200)

The Number of Links Displayed:

Defines the number of links per page, when the navigation option is disabled.

Help by Block:

Indicates if the helps messages from relatives fields are grouped by block. In each field, we can define a help text. With this option activated, it shows up an icon in the block title bar to call the help page.

General Help:

The General Help "consolidates" all the fields help pages in a single page, putting an icon in the toolbar to call the help page.

Toolbar Buttons:

The type of view for the Toolbar buttons (A DIV below the toolbar, Modal).

PDF Settings

Configuração do PDF		
ATRIBUTO	VALOR	DESCRIÇÃO
Orientação do PDF	Retrato	Orientação das páginas dos relatórios gerados em PDF.
Formato do PDF	Carta (216 x 279 mm)	Formato das páginas dos relatórios gerados em PDF.
Tipo de impressão	Ambos	Define o tipo de impressão do PDF.
Gerar PDF Diretamente	<input type="checkbox"/>	Abrir o arquivo PDF gerado sem a necessidade de exibição de uma página intermediária com um link para o arquivo. Opção disponível apenas quando a opção "Criar Gráficos" estiver desabilitada.
PDF Configurável	<input checked="" type="checkbox"/>	Permitir que o usuário configure os parâmetros de criação do PDF durante a execução da aplicação.
Imprimir Background	<input checked="" type="checkbox"/>	Imprimir Background no arquivo PDF.
Tempo de execução do JS	2000	Tempo, em milissegundos, para esperar a execução de JavaScript ao gerar o html para exportação

PDF Orientation

Allows you to set whether to print in Portrait or Landscape orientation.

PDF Format

Allows you to define the type of form in which the PDF will be printed (letter, A4, etc).

Print Type

Allows you to set whether the print mode will be colored or economical.

Generate PDF directly

Opens the generated PDF file without the need to display an intermediate page with a link to the file.

Configurable PDF

Allows the user to configure the PDF creation parameters during the execution of the application.

Print background

Allows you to print a background in the PDF file.

JS Runtime

Maximum time (in seconds) of server waiting when running JS.

Print Settings

Configurações de impressão		
ATRIBUTO	VALOR	DESCRIÇÃO
Tipo de Impressão	Ambos	Define o modo de impressão em relação a cor.
Imprimir Background	<input checked="" type="checkbox"/>	Exibe background na impressão HTML.

Print Mode

Allows you to set the contents of the print file (Both, Current Page, Full Report).

Print background

Lets you set whether the background will be displayed on the printout.

This interface allows configuring the related database settings, such as the Primary Key, Filters, Sorts.

SQL SETTINGS	
ATTRIBUTE	VALUE
Select primary key fields.	
	<div style="display: flex; align-items: center;"> <div style="margin-right: 10px;"> <input type="button" value="On/Off"/> <input type="button" value="All"/> <input type="button" value="None"/> </div> <div style="border: 1px solid gray; padding: 5px;"> <ul style="list-style-type: none"> *orderid customerid employeeid orderdate requireddate shippeddate shipvia freight priceorder shipcountry </div> </div>
Where clause	<input type="text"/>
Order By	<input type="text"/>
Connection	<input type="text" value="conn_mysql"/>
Table Name	<input type="text" value="orders"/>
Variable for Table	<input type="text"/> <input type="text"/>
Case Sensitive	<input checked="" type="checkbox"/>

SQL configuration Interface.

Select primary key fields

It lets you define the Primary key of the Form. ScriptCase already identifies Primary Keys, but you can manually inform or change it by using the buttons beside the fields list. See how the buttons work:

- **n/Off** : Adds or Removes the attribute that defines the primary key for the field. The primary key fields have an asterisk beside their names.
- **ll** : Defines all fields as Primary Keys.
- **one** : Defines none fields as Primary Keys.
- **orting Button** : These are the arrows on the right side of the Combo box. It allows ordering the fields of the Primary Key, placing it in the desired order. To order them, click on the field and use the arrows to move it.

Where clause

It allows adding a WHERE clause to filter the SQL records.

Order By

It allows adding an ORDER BY clause to determine the order to display the records. By default, it uses the primary key to sort the records.

Connection

It allows defining the database connection of the application. You can change the connection to another one that has the same table.

Table Name

It informs the database table used in the Form.

Variable for Table

It allows to use a variable to change a part of the string containing the table name.

Var	<input type="text"/>
orders	<input type="text"/>

Variable for Table Configuration.

Fill the first input with the name of the variable (replace to). The second input you should fill with the part of the name of the table you want to replace (replace from).

Case sensitive

It defines if the database connection uses case sensitive or not.

Field Settings

Sort Fields ?

Select the fields that allow sorting.

On/Off

All

None

*customerid
 *companyname
 *contactname
 *contacttitle
 *birthdate
 *country
 regionid
 stateid
 city
 address

Sort Fields			
customerid	Ascendant ▼	companyname	Ascendant ▼
contactname	Ascendant ▼	contacttitle	Ascendant ▼
birthdate	Ascendant ▼	country	Ascendant ▼
regionid	Ascendant ▼	stateid	Ascendant ▼
city	Ascendant ▼	address	Ascendant ▼
postalcode	Ascendant ▼	phone	Ascendant ▼
fax	Ascendant ▼	cityid	Ascendant ▼
creditlimit	Ascendant ▼	cardtype	Ascendant ▼
cardnumber	Ascendant ▼	notes	Ascendant ▼

Advanced sorting ?

ATTRIBUTE	VALUE	DESCRIPTION
Sorting fields	<input checked="" type="radio"/> Use displayed fields <input type="radio"/> Use columns <input type="radio"/> Define Columns	Default for fields that will be displayed in the advanced sorting
Display icon only on mouseover	<input checked="" type="checkbox"/>	Displays the sorting icon only when the mouse is on the label of the field.

This property defines which fields allow the end-user to sort on the fly, just by clicking on the column (field) he wants to sort by. When sorting the fields, it preserves the GroupBys ordering by default. Also, Scriptcase enables the sorting of the first six application fields, but you can enable it for all fields.

On/Off

All

None

*customerid
 *companyname
 *contactname
 *contacttitle
 *birthdate
 *country
 regionid
 stateid
 city
 address

Sort Fields

It defines the sorting for each field of the application. You can set them as "Ascendant" or "Descendant".

Sort Fields			
customerid	Ascendant ▼	companyname	Ascendant ▼
contactname	Ascendant ▼	contacttitle	Ascendant ▼
birthdate	Descendant ▼	country	Ascendant ▼
regionid	Descendant ▼	stateid	Ascendant ▼
city	Ascendant ▼	address	Ascendant ▼
postalcode	Ascendant ▼	phone	Ascendant ▼
fax	Ascendant ▼	cityid	Descendant ▼
creditlimit	Descendant ▼	cardtype	Ascendant ▼
cardnumber	Ascendant ▼	notes	Ascendant ▼

Advanced Sorting

Define the fields that are available when clicking on the "Sorting" button of the Grid.

With this option, you can insert one or more titles for the columns in the application, allowing to modify its size, color, font, and position.

The Group Label doesn't work with a dynamic display of fields.

The screenshot shows a configuration window titled "Group Label" with a table and a "Cell Properties" dialog. The table has a header row with "Group name" and two data rows: "Title_Summary", "Count_Reg", and "rule1". Below the table are "Add Row" and "Save" buttons. The "Cell Properties" dialog is open, showing fields for Title, Font, Font size, Horizontal alignment, Vertical Alignment, Font Color, and Background Color, with "Update" and "Cancel" buttons at the bottom.

Editing Group Label cell properties

Title

It allows you to define a Title for the Group Label.

Font

Set the font family for the group label title.

Font Size

Set the font size for the group label title.

Horizontal Alignment

Define the horizontal alignment of the group label title. Left, Center, or Right.

Vertical Alignment

Define the vertical alignment of the group label title. Middle, Bottom, or Top.

Font Color

Set the font color for the group label title.

Background Color

Define a background color for the group label title.

This interface allows configuring a Form application to execute the Stored Procedures from your database for Insert, Update and Delete records in the form. You don't need to use the three options simultaneously. If a Procedure option is not enabled, it continues to work as default by using the INSERT, UPDATE and DELETE commands.

See below, the step-by-step to enable a Stored Procedure for inserting records into the database.

Form Procedures ?

Enable INSERT Procedure.

Enable UPDATE Procedure.

Enable DELETE Procedure.

Next

Form Stored Procedures Interface.

- et's check the first option: "Enable INSERT Procedure". Then, inform the Name of the Procedure and the number of parameters.

Form Procedures ?

Enable INSERT Procedure.

Procedure Name

Parameters Qty

Enable UPDATE Procedure.

Enable DELETE Procedure.

Next

Defining parameters for the Procedure.

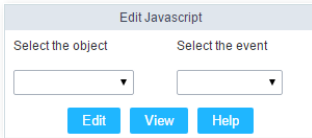
- ow associate the fields for each parameter of the Stored Procedure and the type of parameter (Input or Output). Click on the "Save" button to finish.

Parameters Number	Parameters	Parameters Type	Procedure variable name
Param1	<input type="text"/>	<input checked="" type="radio"/> Input <input type="radio"/> Output	
Param2	<input type="text"/>	<input checked="" type="radio"/> Input <input type="radio"/> Output	
Param3	<input type="text"/>	<input checked="" type="radio"/> Input <input type="radio"/> Output	
Param4	<input type="text"/>	<input checked="" type="radio"/> Input <input type="radio"/> Output	

Save Cancel

Passing the values to the Stored Procedure Parameters.

To coding with JavaScript in Scriptcase, we must associate a JavaScript event to a form field.



Edit JavaScript Interface.

Select the object

This Combo box displays the fields of the form application, and also the form itself as an object

Select the event

Use it to associate the event with the selected field to apply the JS code. View the events available:

- **nClick**: Acts when clicking on the field.
- **nBlur**: Acts when the focus leaves the object.
- **nChange**: Acts when the focus leaves the object, and there are changes in the value.
- **nFocus**: Runs when the focus gets in the object.
- **nMouseOver**: Runs when the mouse pointer hovers the object.
- **nMouseOut**: Runs when the mouse pointer moves out the object.

Events related to the Form

The events below are associated directly with the form object.

- **nLoad**: This event runs when the page is loaded, also when clicking on the navigating buttons.
- **nSubmit**: This event runs when clicking on the “New”, “Save”, and “Delete” buttons.

Edit JavaScript

*Select the object and event, then click on the Edit button. It opens a page to inform custom JavaScript routines and

```
![Edit JavaScript Interface][javascript_edicao_code]
*Edit JavaScript Interface*
```

> The JavaScript language doesn't have the same behavior in all the browsers available. A tip is to test running the

- **nClick Example**
 - hen clicking on a field of type radio, you can enable or disable form fields according to the selected value.

```
if(document.F1.gender[0].checked){
  document.F1.maternity.disabled = false;
  document.F1.maternity.style.background='FFFFFF'
}
if(document.F1.gender[1].checked){
  document.F1.maternity.disabled = true;
  document.F1.maternity.style.background='FCEEB3'
}
```

To access the values of a radio field, you need to use the index.

- **nBlur Example**
 - ou can define a warning for the field “weekly_work_time” when the focus is getting out it.

```
if (document.F1.tp_point[0].checked && document.F1.weekly_work_time.value > '20')
{
  alert("The work time exceeds the limit allowed");
  document.F1.weekly_work_time.value = "";
  document.F1.weekly_work_time.focus();
}
```

- **nChange Example**
 - y modifying the “Salary” of an employee and leaving the field, we’ll check if his “position” is “gardener”.

```
if (document.F1.salary.value > 2000.00 && document.F1.position.value == 'gardener'){
  alert('When I grow up, I want to be a gardener');
}
```

- **nFocus Example**
 - fter informing the purchase value and selecting the payment method in a Select object “Select: pay_method”, the JavaScript code below calculates the value of the purchase.

```
if (document.F1.pay_method[document.F1.pay_method.selectedIndex].text == 'Money')
{
  document.F1.total.value = document.F1.purchase_value.value;
}
```

- **nMouseOver Example**

- You can change the style (background color, font and font color) when the mouse hovers the field.

```
document.F1.field_name.style.backgroundColor = "0FFCCA"
```

- **nMouseOut Example**

- Resets the background color when the mouse moves out from the field.

```
document.F1.field_name.style.backgroundColor = "FFFFFF"
```

This feature allows linking the tables that contain relationships, dependencies, like the tables orders and order_details. This way, it's possible to delete a record in the table orders, and it automatically deletes all the details for that order as well. Below let's see a practical example of this feature.

1. firstly, you must define a new dependency. Then you must select the dependent table. Click on the button New Dependency to start.

Creating a new Dependency.

1. you must inform the fields amount is related between both tables. In this example, we have only one field.

Dependency Table configuration.

1. then select the related key fields: order_details->Orderid - orders->Orderid.

Dependency Key Fields Configurations.

1. now we need to select behaviour for the application. Then click on the button Generate Scripts to define the Dependency Rule.

Configuring dependency Rules.

Security		
ATTRIBUTE	VALUE	DESCRIPTION
Use Security	<input type="checkbox"/>	Application needs validation to run. Read about the sc_apl_status macro.
Url output of the security	<input type="text"/>	Uri to be redirected if the user does not have access to this application.
Use Password	<input type="checkbox"/>	Enter password required to run the application.
Request password just once	<input type="checkbox"/>	Request password just once per session without asking for password every time you open the application in the same session.
Enable direct call by URL	<input checked="" type="checkbox"/>	Allow an application to be called by typing the URL directly in the browsers.

Application Security Configuration

Use Security

Turns the application accessible just when using the macro sc_apl_status or using the security module. When it is enabled, the application displays a message "User not Authorized" if the user tries to access it directly.

Url output of the security

It redirects the application to another URL after displays the warning "User not Authorized".

Use Password

Turns the application accessible just by using a password.

Request password just once

Request the password once per session, not needing to inform the password every time you access the application.

Enable direct call by URL

Allows to call the application directly from the URL in the browsers.

Allows calling the application directly from the URL in the browsers. In the development environment, you can disable those options for testing, it avoids requiring the login every time you run the application. To disable the security and password in development, you must access Options > My ScriptCase and uncheck the options: Enable Use of Security, Enable Use of Password.

Captcha

Captcha		
ATTRIBUTE	VALUE	DESCRIPTION
Use Captcha	<input type="checkbox"/> Captcha	Enables Captcha function for this application.
Number of Characters	<input type="text" value="4"/>	Number of characters
Character List	<input type="text" value="ABCDEFGHIJKLMNOPQRSTUVWXYZ01234"/>	Character list.
Label	<input type="text" value="{lang_othr_cptc_lbel}"/>	Message shown near Captcha.
Error message	<input type="text" value="{lang_othr_cptc_errm}"/>	Captcha error message.
Height	<input type="text" value="65"/>	Picture height in pixels.
Width	<input type="text" value="175"/>	Picture width in pixels.
Font Size	<input type="text" value="20"/>	Font size of the captcha.
Reload	<input type="checkbox"/>	Add a button to reload Captcha.
Select one layout		
<input checked="" type="radio"/> <input type="text" value="cwvQA"/> Label <input type="text" value="cwvQA"/>		
<input type="radio"/> Label <input type="text" value="cwvQA"/> <input type="text" value="cwvQA"/>		

Use Captcha

Defines if the application uses Captcha.

Number of Characters

Amount of characters in the Captcha image.

Character List

List of characters used in the Captcha.

Label

The message displayed for the Captcha.

Error message

Captcha error message.

Height

Height of the image generated by the Captcha (in pixels).

Width

Width of the image generated by the Captcha (in pixels).

Font Size

Size of the font used in the image generated by the Captcha (in pixels).

Reload

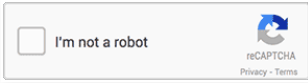
Display the refresh button in the Captcha.

Select one layout

It offers various layouts for the Captcha display.

reCAPTCHA

reCAPTCHA is an API provided by Google for forms. It adds security, preventing automatic submission of forms through robots.

reCAPTCHA sample:

1. first, we must request an API Key to activate reCAPTCHA into a Scriptcase application by following the steps below:

To get a **Site key** and **Secret Key** go to the link: <https://www.google.com/recaptcha/admin#list>. See the image:

Register a new site

Label

For example, example.com: Comments page

Choose the type of reCAPTCHA ?

reCAPTCHA V2
Validate users with the "I'm not a robot" checkbox.

Invisible reCAPTCHA
Validate users in the background.

reCAPTCHA Android
Validate users in your android app.

Domains
(one per line)

For example:
example.com
example.net
example.org

Accept the reCAPTCHA Terms of Service.
By accessing or using the reCAPTCHA APIs, you agree to the Google APIs Terms of Use, and to the Additional Terms below. Please read and understand all applicable terms and policies before accessing the APIs.
» reCAPTCHA Terms of Service

Send alerts to owners ?

Register

Label

It is a project identifier to create the reCAPTCHA keys.

Choose the type of reCaptcha

We must choose the option **reCAPTCHA V2**.

Domains

We can insert multiple domains (one per line) to limit the API uses.

1. then, we need to accept the Terms of Service ("Accept the reCAPTCHA Terms of Service").
2. then clicking on **Register**, the page refreshes and shows the integration of reCAPTCHA information. There we can get the **Site Key** and **Secret Key**:

ⓘ Adding reCAPTCHA to your site

▼ Keys

<p>Site key Use this in the HTML code your site serves to users.</p> <input type="text"/>	<p>Secret key Use this for communication between your site and Google. Be sure to keep it a secret.</p> <input type="text"/>
--	---

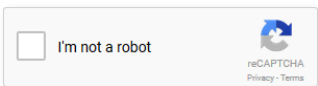
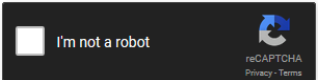
1. now, we can set the Scriptcase application security:

ATTRIBUTE	VALUE
Use Captcha	Recaptcha
Site Key	
Secret Key	
Theme	light
Type	Audio
Size	Compact
Position	Center

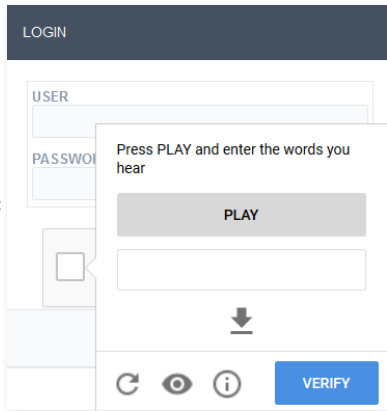
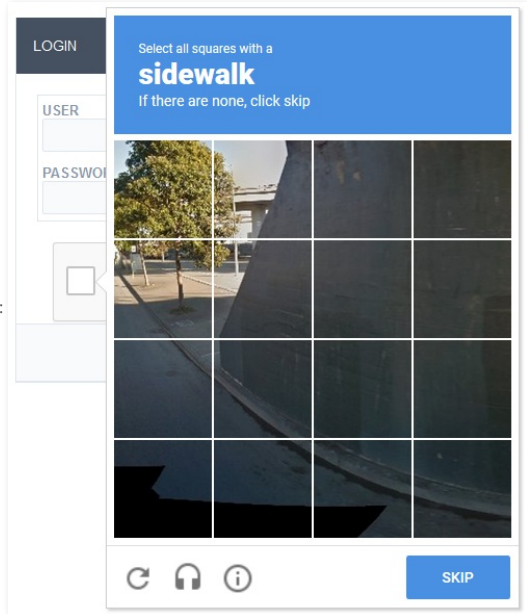
Site Key
Key generated by Google after reCAPTCHA project creation.

Secret Key
Key generated by Google after reCAPTCHA project creation.

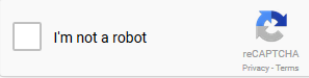
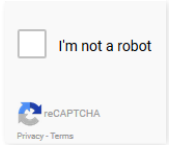
Theme
Defines the reCAPTCHA color. There are two options:

- **light** : 
- **dark** : 

Type
The type of reCAPTCHA. There are two options:

- **audio**: 
- **image**: 

Size
The size of reCAPTCHA. There are two options:

- **normal**:  A rectangular widget with a light gray background. On the left, there is a small square checkbox followed by the text "I'm not a robot". On the right, there is a circular reCAPTCHA logo with the text "reCAPTCHA" and "Privacy - Terms" below it.
- **compact**:  A rectangular widget with a light gray background. On the left, there is a small square checkbox followed by the text "I'm not a robot". On the right, there is a circular reCAPTCHA logo with the text "reCAPTCHA" and "Privacy - Terms" below it.

Position

Here we can define the reCAPTCHA component alignment:

- **eft**: Position the reCAPTCHA component at the left.
- **enter**: Position the reCAPTCHA component at the center.
- **ight**: Position the reCAPTCHA component at the right.

This interface allows you to define a Log schema to the app. The Log schema tracks what the end-users are doing in the app. Those are the events it can tracks: Access, Insert, Update, and delete.

To learn how to create a Log schema [see here](#).

The screenshot shows a configuration window titled "LOG". It features a table with two columns: "ATTRIBUTE" and "VALUE". The "Schema" attribute has a dropdown menu currently set to "log". Below the table, there are two empty list boxes: "Events" on the left and "Access" on the right. Between these two boxes are two blue buttons: "Add >>" and "Remove <<".

Application Log configuration

Schema Select an existing log schema for the application.

Events You can select just specific events to save in the log.

1. [Home](#)
2. Applications

General Overview

The applications can have two types of fields.

Table Fields

Columns of the table connected to the application and their data types.

Example:

In a table with the following structure:

```
CREATE TABLE `orders` (
  `orderid` int(11) NOT NULL,
  `customerid` varchar(5) NULL DEFAULT NULL,
  `employeeid` int(11) NULL DEFAULT NULL,
  `orderdate` date NULL DEFAULT NULL,
  `requireddate` date NULL DEFAULT NULL,
  PRIMARY KEY (`orderid`) USING BTREE
);
```

These are the listed fields on left menu **Fields**:

How to synchronize table fields with a form

Anytime you add/delete a field or change its data type, you have to synchronize your form application with the connected table to apply that changes.

For more information, click [here](#).

Virtual Fields

Fields created inside application to assist in the development process.

How to create a virtual field

1. Inside an application, access the left menu **Fields** and click on **New Field** option.
2. Choose how many fields do you want to create:

1. Create the fields and define type, name and label for each one:

- **Type:** Data type of the field.
- **Name:** Internal name of the field. Used to identify the field on events and Scriptcase interface.
- **Label:** Title of the field displayed on the application.

Scriptcase **doesn't insert** virtual fields on the table after its creation.

Types of fields by applications

Application Table Fields Virtual Fields

Grid	<input type="checkbox"/>	<input type="checkbox"/>
Procedure	<input type="checkbox"/>	<input type="checkbox"/>
Chart	<input type="checkbox"/>	<input type="checkbox"/>
Form	<input type="checkbox"/>	<input type="checkbox"/>
Control	<input type="checkbox"/>	<input type="checkbox"/>
Search	<input type="checkbox"/>	<input type="checkbox"/>
PDF Report	<input type="checkbox"/>	<input type="checkbox"/>
Calendar	<input type="checkbox"/>	<input type="checkbox"/>

The applications **Menu**, **Tree Menu**, **Dashboard** and **Blank** were not listed because they haven't manage fields.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Form
- 4. Text

Text

General Settings

This type of field allows the developer to create quickly inputs to insert and update data, where the final user can inform its data to be allocated in its database.

Type of Data

Defines the type of the field in the application. In this case we should select **Text**.

Label

Defines the title that will be displayed in the field when the application is executed. The terminology used in the interface is fundamental to the system have a good usability, who should use names and familiar terms to the final user of the application, instead of using terms from the system.

For example, this text field that has the name **customerid**, the client would have a much better understanding of the functionality of the field when we define the label as **Customer Name**.

Not only a fixed text, the **Label** attribute allows the use of langs to define the field's title, making it possible to use your application in a multi language project.

Watermark

Informing a text to the **Watermark** it will be displayed in the input a text as an example that can be informed in the field. The result after set will be this:

Initial Value(type)

Allows the initial definition to the field when the form is in insert mode. It is possible to choose between two options:

Defined Value: When this option is selected, the Initial Value attribute will be available, where we should inform the field's initial value. For example, my initial value is **Arlindo**, when a new register is inserted, the field **Seller Name** will be initialized as Arlindo.

System Data: When this option is selected, the initial value will be the actual date of your computer's system.

Amount of Characters

Allows to set the width of the text field's input that varies with the amount of characters informed. Although, if the amount of characters typed are greater than the setting, the text will be pushed to the left, to keep the maximum amount of characters as defined.

Show HTML Content

When this option is active every HTML, CSS and JavaScript content that are in the database will be displayed with the main value.

Validation Image

When this option is active, a image will be displayed next to the informed field if the field is according to the settings defined by the developer.

In the example below, the field was set to receive at least 5 characters, see what happens when informed only 4 characters:

However if informed 5 or more characters the field will be displayed as:

Password Field

When this option is active, the text field will be converted to the format used in password fields. For example:

Save Variable

Allows to save a session variable(global variable) with the field's value, to be used in others applications.

For example, in the login form the username can be saved in session and displayed on the header of others applications.

Variable Name

In this attribute we should define the name of the session variable, active in the previous item, that will receive the field's value.

We should inform only the variable's name, - **var_rating**.

The method to use its value is [global variable](#).

Field Mask

Defines the field mask. There are two types of mask described in the table below:

Character Description

- | | |
|---|--|
| X | It is overridden by any characters returned by the database. Filled is required and the value will be completed with leading zeros when there are fewer characters than the size of the mask. |
| Z | It is overridden by any characters returned by the database. Its filling is optional and when there are fewer characters than the mask size, nothing will be done about the characters that are missing. In addition the leading zeros will be suppressed. When used in conjunction with the X necessarily the use of this character is to the left of the mask. |
| 9 | It represents any numeric character (from 0-9) |
| A | It represents an alpha numeric character (A-Z,a-z) |
| * | It represents any alpha-numeric character (A-Z,a-z,0-9)entered by the user. |

For example, it is possible to set the mast to display a telephone number:

It will be show with this format on runtime:

It is also possible to set the field mask like those examples:

Field mask examples:

Telephone number

Field	Mask Input	Typed Value	Formatted Value
Phone Number	+99 99 9999 - 9999	123456789012	+12 34 5678 - 9012
Phone Number	(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
Phone Number	(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
Phone Number	(zz) xxxx - xxxx	12345678	() 1234 - 5678
Software Key (Letters only)	AAAA-AAAA-AAAA-AAAA	QWERTYUIASDFGHJK	QWER-TYUI-ASDF-GHJK
Software Key (Letters and numbers)	xxxx-xxxx-xxxx-xxxx	Q1W2E3R4T5Y6U7I8	Q1W2-E3R4-T5Y6-U7I8
License Plate	AAA - 9999	QWE1234	QWE - 1234
Scriptcase Serial Key	A999A999A999-*	D111H111G111DG2P	D111H111G111-DG2P
Multiple masks (Phone Number)	9999-9999;(99)9999-9999; 9999 999 9999		+99 99 9999-9999

Validate with mask

When this option is active it is possible to insert the data with the mask set in the **Field Mask** option.

Complete to the left

When this option is used, the value defined will be added on the left of the information inserted in the field if the value is lower than the maximum set in the **Maximum Size** option.

Field size in database

Defines the field's size related to the size set in the database. This value is already set automatically by default when the application is generated.

Hidden Field

When this option is active, the field will be hidden in the application on runtime.

Label Field

When this option is active, the field will be altered to only a label where the info will be displayed, where updates or inserts will not be possible.

Save HTML tags

When this option is active, it allows to HTML tags in the field to be inserted with the data, instead of being interpreted.

Text input in JavaScript

When this option is active, it will be show every JavaScript content inserted in the database with its main value.

This option can only be used using the **Editable grid view**.

Disable Field

Allows the developer to disable the field, so the user can not type a value according with the option defined by the developer.

The available options are: **No** - Initial value of the attribute, this option does not disable the field.

Update Mode - This option only disables the field when editing existing registers.

Insert Mode - This option only disables the field when inserting new registers.

Insert / Update Mode - This option disables the field when editing exciting registers or when inserting new registers.

HTML Type

HTML object used to display the field in the form.

SQL Type

Informs the type of the field in the database.

Values Format

Format of Values Interface

- **Case Settings** : Allows to convert the letters of the field when losing the focus. The options are:
 - **Upper Case** : All in Upper Case.
 - **Lower Case** : All in Lower Case.
 - **Capitalize first word** : Capitalizes the first letter of the word.
 - **Capitalize all words** : Capitalizes the first letter of all the words.
- **Minimum Size** : Allows to define the minimum amount of characters accepted for the field.
- **Maximum Size** : Allows to define the maximum amount of characters accepted for the field.
- **Allowed Characters** : Allows to select a set of characters that can be typed into the field. The options are:
 - **All** : Allows any type of character to be typed into the field
 - **Selected** : Defines a set of characters allowed. Using the configuration below, will be allowed to type letters, numbers and other characters (defined in the attribute More). The letters a, b and c (defined in the Less Attribute) are not allowed.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. Multiple Lines Text

Multiple Lines Text

General Settings

Multiple Lines Text field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Multiple Lines Text , you can inform a Text value to the field in multiple lines.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : you will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Lines** : Allows to define the amount of lines that the field will have at start.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Field size in database** : Determines the field size in bytes. it is used to determine the max size of characters that is allowed to type in.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Save HTML tags** : This option allows to save the HTML content of the field to the database.
- **Text input in JavaScript** : Allows to save JavaScript code informed in the field to the database.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface

- **Case Settings** : Allows to convert the letters of the field when losing the focus. The options are:
 - **Upper Case** : All in Upper Case.
 - **Lower Case** : All in Lower Case.
 - **Capitalize first word** : Capitalizes the first letter of the word.
 - **Capitalize all words** : Capitalizes the first letter of all the words.
- **Minimum Size** : Allows to define the minimum amount of characters accepted for the field.
- **Maximum Size** : Allows to define the maximum amount of characters accepted for the field.
- **Allowed Characters** : Allows to select a set of characters that can be typed into the field. The options are:
 - **All** : Allows any type of character to be typed into the field
 - **Selected** : Defines a set of characters allowed. Using the configuration below, will be allowed to type letters, numbers and other characters (defined in the attribute More). The letters a, b and c (defined in the Less Attribute) are not allowed.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.

- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Form
- 4. Integer

Integer

General Settings

Data type:

It sets the application field type. When the field is set as Number, it is permissible to define formatting rules of integers.

Label:

It sets the title that will be displayed in the field when you run the application. The terminology used in the interface is fundamental for your system has a good usability. In this case, we recommend you to use names and terms familiar to the end user of the application, instead of using terms originated in the system.

Watermark

Informing a text for the **watermark** it will be displayed on the input text with an example of what can be reported in the field. The result after that will be configured:

Initial value (type)

Allows setting an initial value for the field when the application is in inclusion. You can choose between two options:

Pre-set value: When you select this option, the Initial Value attribute will be made available in the place where we inform the initial value of the field. For example, my initial value is **Arlindo** when inserting a new record the **seller_name** field will start with Arlindo.

System date: When you select this option, the initial value will be the current system date of your computer.

If you select the type system date, it will be not necessary to fill in the initial value attribute.

Amount of characters

It allows you to set the width of the input text field according to the amount of characters. However, if the quantity entered is greater than the set for the characters, the text will be pushed to the left, in order to ensure the maximum amount of characters set in the option of **Values formatting**.

Validation Image

When you enable this option, an image will be shown next to the field informing whether the field is in accordance with the settings of *minimum size* and *Maximum size* (as you can see in the images below) defined in the Formatting value option.

In the example below, the field was set to receive at least 5 characters, see what happens with the image when it receives only 4 characters:

However, if the value inserted has 5 or more characters the the image will change according to example bellow:

Use slider:

It displays a slider component in the field. So you can increase or decrease the value sliding the cursor. You can also customize the increment value, if it increments the value 1 by 1, 2, 5, 10... N.

Field Mask Input:

It sets the mask input for the field. There are two types of mask as described in the table below:

Character Description

X	It is overridden by any characters returned by the database. Filled is required and the value will be completed with leading zeros when there are fewer characters than the size of the mask.
Z	It is overridden by any characters returned by the database. Its filling is optional and when there are fewer characters than the mask size, nothing will be done about the characters that are missing. In addition the leading zeros will be suppressed. When used in conjunction with the X necessarily the use of this character is to the left of the mask.
9	It represents any numeric character (from 0-9)
A	It represents an alpha numeric character (A-Z,a-z)
*	It represents any alpha-numeric character (A-Z,a-z,0-9)entered by the user.

Mask Input Examples:

Field	Mask Input	Typed Value	Formatted Value
Phone Number	+99 99 9999 - 9999	123456789012	+12 34 5678 - 9012
Phone Number	(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
Phone Number	(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
Phone Number	(zz) xxxx - xxxx	12345678	() 1234 - 5678
Software Key (Letters only)	AAAA-AAAA-AAAA-AAAA	QWERTYUIASDFGHIJK	QWER-TYUI-ASDF-GHIJK
Software Key (Letters and numbers)	xxxx-xxxx-xxxx-xxxx	Q1W2E3R4T5Y6U7I8	Q1W2-E3R4-T5Y6-U7I8
License Plate	AAA - 9999	QWE1234	QWE - 1234
Scriptcase Serial Key	A999A999A999-**	D111H111G111DG2P	D111H111G111-DG2P
Multiple masks (Phone Number)	9999-9999;(99)9999-9999; 9999 999 9999		+99 99 9999-9999

Validate with Mask

By enabling this option you will be able to enter data according to the mask that was configured in the option **Field Mask** and Scriptcase will validate it.

Record Variable

Allows you to record a session variable with the value of the field ([global variable] [var_glob]), to be used in other applications.

Example, in the login form the username can be saved in session and displayed in the header of other applications.

Variable Name

In this option you must set the session variable name, enabled in the previous item, which will receive the value of the field.

You need to inform only the variable name, for example: **var_seller**.

The recovery of the value is made as [Global Variable][var_glob]{:target='blank'}.

Field size in Database

This option sets the size of the field relative to the size that is configured in the database. By default this value is already configured automatically when the application is generated.

Hidden Field

This option when enabled will hide the field inside the application at the time of execution.

Label Field

By enabling this option, the field will be changed to only one label where the information will be displayed, so it is not possible to make changes or inserts in the field configured as label.

Disable field

Allows you to disable the field, making it impossible for the user to enter a value according to the developer-defined option.

The available options are:

No - initial value of the attribute, this option does not disable the field. **Update mode** - This option disables the field only when editing the records. **Insert mode** - This option disables the field only inserting new records. **Update/Insert mode** - This option disables the field in both editing and inserting new records.

HTML Type

HTML object used to display the field in the application.

Tipo SQL:

Reports the SQL type of the field as it is configured in the database.

Values Format

Use regional settings:

It allows you to apply regional settings to the number fields formatting. When this option is not selected it will be displayed the attributes group, minus sign and negative number format.

Digit grouping:

It allows you to define if the field displays the digit grouping separator.

Maximum Size:

It allows you to define the maximum size that will be used in the field of the application.

Minimum Value:

It allows you to define the minimum value that will be used in the field of the application.

Maximum Value:

It allows you to define the maximum value that will be used in the field of the application.

Grouping:

It allows you to define which character is used to separate thousands. This option is only available when the option to use regional settings is disabled.

Negative sign:

It allows you to define which character will be used for the display of negative numbers. This option is only available when the option to use regional settings is disabled.

Negative number format:

It allows you to define the placement of the negative sign in relation to value. This option is only available when the option to use regional settings is disabled.

Accept:

It allows to define if the field accept only negative values, positive-only or both.

Display Calculator:

This option when enabled will display a calculator to assist the end user to perform calculations.

Use Spin:

This option applies a Spin component to the number field. So the end user can increase or decrease the numbers using it. You will also be able to set a minor or major range for the spin.

Smaller increment for the Spin::

Sets the minimum range of the Spin component.

Greater increment for the Spin:

Sets the maximum range of the Spin.

Database Value

It allows you to define how the value to be written to the database.

Preset value:

It allows you to specify a fixed value or associate it to global variables created inside the project and save it automatically to the database field.

Auto increment (automatic):

It allows you to use the increase generated automatically by the database to fill the field. It is used only when the database field is of type AUTO INCREMENT or similar. For the databases that use string, such as Oracle, PostGres and Firebird, you must inform the name of the string.

Auto increment (manual):

The application itself will simulate an auto increment in the field. So the value of the field will be calculated automatically by the application.

Date of inclusion:

The field will be populated automatically with server date during the inclusion.

Date of update:

The field will be populated automatically with server date during the update.

User IP:

The field will get the IP of the machine that is accessing the application.

Calculated by the database:

The field value is assigned by the database itself. The field will not be used in the process of updating. For example: it will be updated via a Trigger.

Null:

The field will be automatically filled with null.

Lookup Settings

The Lookup is a way used to provide the end user a list of values description/name where the items can be selected and used in the application. For example, for a field sex that in the database is represented with M or F values can be presented as male or female dynamically with the lookup. So you will be able to retrieve these values from a database table or manually, if the reference table does not exist.

Example:

Lookup Methods

There are two lookup methods available:

Manual: To use this method it is necessary to inform the selection conditions manually to the Lookup.

Automatic: To use this method it is necessary to use a lookup query from the database to access the reference table.

Editing-automatic lookup:

Select Command:

It sets the SQL command that will retrieve the value to display inside the Grid field. The command must have the following format

```
SELECT field_to_be_displayed FROM table WHERE key_field = {grid_field}
```

The Grid field must be always referenced between curly braces {}, So at run time, the field braces will be replaced by its value in the table.

Multiple Options:

When the Select command informed return multiple values, the option must be selected.

Delimiter:

It sets the tab of values returned by the Select command, this option should be filled when the "Yes" option is checked in the multiple attribute options.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

Use a specific connection:

This option allows you to select another existing connection in the project, other than the current connection, to select the values for the lookup.

Edit Lookup - Manual:

Lookup used to modify the display of the field with predefined values.

Lookup Types

It sets the operating mode of the lookup can be: single value, multiple values (delimiter), multiple values (position) and multiple values (binary).

Single value: it will be displayed only one value for the field. There is a direct replacement of the value stored in the Bank for a label defined in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value of "M" will be replaced by "Male".

Multiple Values (Delimiter):

By using this option it will be displayed multiple values for the selected field. The values must be separated by a delimiter to be informed. A replacement of parts from a string stored in the field, separated by a delimiter for values contained in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value S;M will be replaced by Sports and Movies.

Delimiter: Character used to separate the values within the same string.

Multiple Values (Position)

Allows you to recover information stored in a single string of the selected field. In order for this information to be retrieved must be informed, in addition to the label, the starting position and the number of bytes that each information occupies within the string.

As an example we will use a string to store Sex, Marital status and Hobby respectively. Sex occupies one byte, Status one byte and Hobby two bytes .

For this purpose we define the list as:

Label	Value	Start	Size
Male	M	1	1
Female	F	1	1
Married	M	2	1
Single	S	2	1
Study	ST	3	2
Sportes	SP	3	2
Reading	LE	3	2

Example: the string MMST would be presented in the query as: Male Married Study

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value M will be replaced by Male.

Start: : Starting position in the string where the information is recorded. The first position is always 1.

Size: : Number of bytes that the information occupies in the string.

Multiple values (binary):

It allows you to retrieve several information stored in decimal form of the selected field.

As examples we will use the following list (although not informed in the interface of inclusion of values in the list, each item has a value assigned automatically according to the order):

Assigned value Description in Lookup

1	Sports
2	Culture

Assigned value Description in Lookup

8	Reading
16	Music

To display the data, a decomposition of the decimal number stored in the bank is performed. For example, numbers 11 and 12 (written in the database) would be decomposed as follows:

11 = 1 + 2 + 8 = (Sports - Culture - Reading)

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

12 = 4 + 8 = (Leisure - Reading)

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

Label:

Text that will be displayed in the Grid.

Include Button:

Includes in the list the values populated in the Label and value fields.

Alter Button:

It changes the attributes of the selected item.

Delete Button:

Deletes the selected item.

Clean Button:

It cleans the fields

Save Button:

Allows the user to save the entire list, for later use using the Load Lookup definition option.

Load Lookup definition:

Allows you to load a predefined value list for use as a lookup.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

• **CSS of the Title**

- **Font** : Allows to choose the font type, that will be applied to the application field title.
- **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.

- **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. Decimal

Decimal

General Settings

Decimal field configuration Interface.

- **Data Type** : Select the type of field for the application. When set to Decimal, you can inform decimal values to the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Defines the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Defines the amount of characters allowed for the field.
- **Validation Image** : Displays an image next to the field when it's being validated.
- **Save Variable** : Saves a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Defines the name for the session variable that'll receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Applies the Regional Settings to format the number of the field. When not enabled, the attributes Digit Grouping Separator, Decimal Separator, Negative sign and Negative number format.
- **Digit grouping** : Defines if the field will display the digits separator.
- **Maximum Size** : Determines the max size of the field.
- **Minimum Value** : Set the min value of the field.
- **Maximum Value** : Determines the max value of the field.
- **Decimal Precision** : Defines the amount of decimal places for the field.
- **Digit Grouping Separator** : Defines the character that is used to separate the thousand.
- **Decimal Separator** : Defines the decimal separator.
- **Negative sign** : Defines the character that will be used to display negative numbers.
- **Negative number format** : Defines the position of the Negative sign em relation to the value.
- **Accept** : Determines if the field, will accept only negative, positive or both numbers.
- **Show a calculator** : Displays an icon next to the field, which allows the user to calculate a value for the field.
- **Auto-complete with zeros** : If enabled, it automatically informs the decimal places when not specified. If disabled, the user needs to always inform the decimal value.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced by the curly brackets {}. While running the application, the field within the curly brackets is replaced by its value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Form
- 4. Currency

Currency

General Settings

Currency field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Currency, you can currency values to the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through Javascript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the datatype of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the number of the field. When not enabled, the attributes Digit Grouping Separator, Decimal Separator, Negative sign and Negative number format.
- **Digit grouping** : Defines if the field will display the digits separator.
- **Currency symbol usage** : Defines if the field will display the Currency Symbol of the Regional Settings.
- **Maximum Size** : Determines the max size of the field.
- **Minimum Value** : Determines the min value of the field.
- **Maximum Value** : Determines the max value of the field.
- **Decimal Precision** : Defines the amount of decimal places for the field.
- **Digit Grouping Separator** : Defines the character that is used to separate the thousand.
- **Decimal Separator** : Defines the decimal separator.
- **Negative sign** : Defines the character that will be used to display negative numbers.
- **Negative number format** : Defines the position of the Negative sign em relation to the value.
- **Accept** : Determines if the field, will accept only negative, positive or both numbers.
- **Show a calculator** : Displays an icon next to the field, which aids the user to calculate a value for the field.
- **Auto-complete with zeros** : If enabled, it automatically informs the decimal places when not specified. If disabled, the user needs to always inform the decimal value.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGRES and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced by the curly brackets {}. While running the application, the field within the curly brackets is replaced by its value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. Date

Date

General Settings

Date field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Date, you can inform a date.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows it is value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator attribute.
- **Date separator** : Allows you to inform the separator symbol for the date.
- **First Day** : Define the first day of the week.
- **Display** : Select the format of the day for display.
- **Internal Format** : Allows to define the format the field when the SQL type is different from **Date**. You need to use the characters **Y**, **M** and **D** that correspond to **Year**, **Month** and **Day**.
- **Use Combo-box** : Allows you to select the date using a combo-box.
- **Minimum Date** :
 - **Fixed Date** : Inform the minimum date allowed presented in the selected date format.
 - **Current Date** : Clicking on the icon next to the text field, ScriptCase offers the following options:
 - **Actual date** : It will place the current date as the max allowed date.
 - **Actual date with increment** : The minimum date will be the current date (+) the days or months or years that you want to increment.
 - **Actual date with decrement** : The minimum date will be the current date (-) the days or months or years that you want to decrement.
- **Maximum Date** :
 - **Fixed Date** : Inform the minimum date allowed presented in the selected date format.
 - **Current Date** : Clicking on the icon next to the text field, ScriptCase offers the following options:
 - **Actual date** : It will place the current date as the max allowed date.
 - **Actual date with increment** : The minimum date will be the current date (+) the days or months or years that you want to increment.
 - **Actual date with decrement** : The minimum date will be the current date (-) the days or months or years that you want to decrement.
- **Display Format** : Enables the Date format beside the field when informing the date.
- **Display Calendar** : Enables the a calendar icon beside the field, this allows to select the date from a calendar with the format already setup.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGRES and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Form
- 4. Time

Time

General Settings

Time field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Time, you can inform a time to this field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator and time separator attributes.
- **Time separator** : Allows you to inform the separator symbol for the time.
- **Display** : Select the format of the time for display.
- **Internal Format** : Allows to define the format the field when the SQL type is different from **TIME**. You need to use the characters **HH**, **II**, and **SS** that correspond to **Day**, **Hour**, **Minutes** and **Seconds**.
- **Display Format** : Enables the Time format beside the field when informing the time.
- **Use Time picker** : Use the JQuery plug-in to choose the time.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets `{}`. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.

- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Form
- 4. Date and Hour

Date and Hour

General Settings

Date and Time field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Datetime, you can inform a date and time to this field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows it is value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator and time separator attributes.
- **Date separator** : Allows you to inform the separator symbol for the date.
- **Time separator** : Allows you to inform the separator symbol for the time.
- **Display** : Select the format of the day/time for display.
- **Internal Format** : Allows to define the format the field when the SQL type is different from **DATETIME**. You need to use the characters **Y, M, D, HH, II, and SS** that correspond to **Year, Month, Day, Hour, Minutes and Seconds**.
- **Display Format** : Enables the Date and Time format beside the field when informing the date.
- **Group Date and Time** : Groups the date and time in the same field.
- **Display Calendar** : Enables the a calendar icon beside the field, this allows to select the date from a calendar with the format already setup.
- **New Calendar** : Show the new jquery calendar, or the old calendar.
- **Years Limit** : The amount of years that will display in the calendar.
- **View week number** : Display the week number in the calendar.
- **Additional months** : Display additional months in the calendar.
- **Show Combo year and month** : Allows you to select the year and month using a combo-box.
- **Decimal of seconds** : Defines the amount of decimals for the seconds.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.** : When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced by the curly brackets {}. While running the application, the field within the curly brackets is replaced by its value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. HTML Image

HTML Image

General Settings

HTML Image field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as HTML Image, it allows to place an image to the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Image** : Defines an image that will be displayed. The icon "Select Image", lists all the standard images from scriptcase and also the image that you have uploaded to scriptcase. The icon "Upload an image" allows the developer to send an image to the server which is from another machine.
- **Border** : Border size for the image in Pixels.
- **Width** : Define the width of the image in Pixels.
- **Height** : Define the height of the image in Pixels.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. Label

Label

General Settings

Label field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as label, the field only display the output of the value .
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Text** : You can inform the text that will display beside the field.
- **Reload** : When enabled, make the text informed for the field to become a link to reload the form when clicked.
- **Position** : Allows to position the field in the chosen block.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. Card

Card

General Settings

Credit Card Number Configuration Interface.

- **Data Type** : Defines the type of field for the application. When it is defined as a Credit Card Number, the field verifies if the value is valid.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name would be "Client Name".
- **Watermark** : Creates a placeholder on the field with the text informed.
- **Initial Value (type)** : Set the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Defines the amount of characters allowed for the field.
- **Validation Image** : Display an image next to the field when it is being validated.
- **Save Variable** : Saves a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Set the name for the session variable that will receive the field value.
- **Field size in database** : Determines the size in bytes of the fields. It is used to determine the max amount of characters to be typed in.
- **Field Mask** : Defines the display mask for the field. There are three types of masks that can be merged.

Character	Description
9	Represents a numeric character (0-9)
A	Represents an alpha numeric character (A-Z,a-z)
*	Represents any alpha numeric character (A-Z,a-z,0-9) typed by the user.

It is possible to merge two or more masks simultaneously, separated by a semi coma with the smallest mask at start. The replacement occurs when the user is typing when the lowest amount of character exceeds.

Examples of Masks

Field	Mask	Informed Value	Formatted Value
Telephone Number	+99 99 9999 - 9999	123456789012	+12 34 5678 - 9012
Software Key (Only Numbers)	AAAA-AAAA-AAAA-AAAA	QWERTYUIASDFGHJK	QWER-TYUI-ASDF-GHJK
Software Key (Letters an Numbers)	**_**_**	Q1W2E3R4T5Y6U7I8	Q1W2-E3R4-T5Y6-U7I8
License Plates	AAA - 9999	QWE1234	QWE - 1234
ScriptCase License	A999A999A999.**	D111H111G111DG2P	D111H111G111-DG2P
Multiple Masks (Telephone)	9999-9999;(99)9999-9999;	9999 999 9999	+99 99 9999-9999

- **Hidden Field** : This option makes the field hidden, but still allows it is value to be processed through JavaScript or PHP.
- **Label Field** : This option make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Display the data type of field in the database.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGRES and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. Card type

Card type

General Settings

Credit Card field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Credit Card , it'll setup a combo-box for you to select a creditcard type.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it'll be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that'll receive the field value.
- **Field size in database** : Determines the field size in bytes. It's used to determine the max size of characters that's allowed to type in.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that'll be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface.

- **Credit Card** : Allows to define which cards will be listed and their internal value.
- **Use Title/Deny** : Allows to inform a new value if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen. The label and the value of this option are defined as the Negative Value and the Title.
- **Negative value** : Value of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Title** : Label of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Related Field** : You'll inform which of the fields contains the values of a Credit Card Number so that the validations can be done correctly.
- **Reload Form** : Reloads the form when the field's value is modified.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGRES and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Date of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookups Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. E-mail

E-mail

General Settings

Email field configuration Interface.

- **Data Type** : Defines the type of field for the application. When set to Email, the field applies validations for an email format.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Defines the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Defines the amount of characters allowed for the field.
- **Validation Image** : Displays an image next to the field when it is being validated.
- **Save Variable** : Saves a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Defines the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through Javascript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Display Icon** : Displays the Email icon next to the field.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface.

- **Credit Card** : Allows to define which cards will be listed and their internal value.
- **Use Title/Deny** : Allows to inform a new value if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen. The label and the value of this option are defined as the Negative Value and the Title.
- **Negative value** : Value of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Title** : Label of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Related Field** : You'll inform which of the fields contains the values of a Credit Card Number so that the validations can be done correctly.
- **Reload Form** : Reloads the form when the field's value is modified.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.

- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. URL

URL

General Settings

URL field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to URL, you can inform a Link value to the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Display Icon** : Displays an icon beside the field.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **Show the URL as a clickable link** : Sets the content of the field to clickable link.
- **Target handling where the link will open** : Target where the the lick will direct to after being clicked
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface.

- **Credit Card** : Allows to define which cards will be listed and their internal value.
- **Use Title/Deny** : Allows to inform a new value if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen. The label and the value of this option are defined as the Negative Value and the Title.
- **Negative value** : Value of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Title** : Label of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Related Field** : You'll inform which of the fields contains the values of a Credit Card Number so that the validations can be done correctly.
- **Reload Form** : Reloads the form when the field's value is modified.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.** : When enabled, the lookup settings is enabled and opens more settings.

- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.

- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. HTML Color

HTML Color

General Settings

HTML Color field configuration Interface.

- **Data Type** : Select the type of field for the application. When set to HTML Color, you can select a color to be used in the form.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Initial Value (type)** : Set the initial value to the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Defines the amount of characters allowed for the field.
- **Validation Image** : Displays an image next to the field when it is being validated.
- **Save Variable** : Save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Defines the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of the field in the database.

Values Format

Format of Values Interface.

- **Credit Card** : Allows to define which cards will be listed and their internal value.
- **Use Title/Deny** : Allows to inform a new value if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen. The label and the value of this option are defined as the Negative Value and the Title.
- **Negative value** : Value of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Title** : Label of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Related Field** : You'll inform which of the fields contains the values of a Credit Card Number so that the validations can be done correctly.
- **Reload Form** : Reloads the form when the field's value is modified.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGRES and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.** : When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:


Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. HTML Editor


HTML Editor

General Settings

 HTML Editor field configuration Interface.

- **Data Type** : Defines the type of field for the application. When set to HTML Editor, you can inform any type of character and they will be saved in HTML form.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Height** : Set the Height in pixels to the HTML Editor field.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through Javascript or PHP.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **SQL Type** : Displays the data type of field in the database.


Toolbar

 HTML Editor toolbar settings Interface.


- **Properties**
 - **Position** : Location of the HTML Editor toolbar.
 - **Alignment** : Toolbar button alignment.
 - **Status** : Status Bar Display (Do not Display, Top and Bottom).
 - **Amount** : The Amount of lines of the HTML Editor toolbar.
- **Button Organization** : Positions the toolbar buttons.
- **Preview** : You can visualize the toolbar according to the previous settings.

Database Value


Allows to define a value to be saved to the database.

 Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

 Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

 Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.

- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the Input Object.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
- **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the Input Object.
- **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. Localization

Localization

General Settings

Location field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it's defined as Location, you will see a combobox with the Languages that are part of the Project Properties.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Reload** : When enabled, make the text informed for the field to become a link to reload the form when clicked.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Form
- 4. Theme

Theme

General Settings

Theme field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Theme, you view a combo box with the list of themes that are set to your project.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Reload** : Reload the form after modifying the value.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

-
- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
 - **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
 - **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. Youtube

Youtube

General Settings

YouTube field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a YouTube, it allows to display a video from youtube on the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Display Mode** : Defines how the you video will display in the application.
- **Width** : Width of the video in pixels.
- **Height** : Height of the video in pixels.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

-
- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
 - **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
 - **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS

Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Form
- 4. Google Maps

Google Maps

General Settings

Google Maps field Configuration Interface.

- **Data Type** : Select the type of field for the application. When it is defined as Google Maps, it will use the Google Maps API to display the map in the Form Applications
- **Label** : Set the title to the field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Home** : Defines what type of parameters will be used in the API.

Home Configuration Interface.

- **Display Mode** : Indicates the display mode of the map. It can be opened in a Modal or in a new Window.
- **Width** : Defines the width of the map that is going to be displayed.
- **Height** : Defines the height of the map that is going to be displayed.
- **Zoom** : Defines the initial Zoom (available from the Google API) of the Map location.
- **API Key** : API Key for authorization to use Google Maps in the Application. (Required only for the versions 2 or earlier of the Google API.)

The API Key is a unique key, composed by a string(text) alphanumeric, which is the license to use the service. When you subscribe to use the service, the key is tied to the domain and the directory of the server. All the pages that use the API needs to be in the same directory that was used for the subscription. In case you have a web server on your local machine, you just need to possess a key for testing, and to do this you only need to set http://localhost in the domain of the subscription.

To get your API Key access the site by [clicking here](#)

- **Link Type** : Defines how the link will be displayed.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. Image (Database)

Image (Database)

General Settings

Image (Database) field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as an Image (Database), all the Images files are stored and loaded directly from the Database.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Image Border** : Width of the Image border. Value in Pixels.
- **Image Height** : Image height size. Value in Pixels.
- **Image Width** : Image width size. Value in Pixels.
- **Progress bar** : Displays a progress bar when sending the files to the server.
- **Upload area** : Displays a drag and drop area to upload the file.
- **Maintain Aspect** : Maintains the original aspect ratio of the image when resizing it.
- **Display link only** : When enabled, you will be presented with a link to open the image in another image with the original size of the image (without using the re-dimensioning).
- **Open in Another Window** : Allows to open the image in another window.
- **Extensions to allow** : Extensions allowed to upload, separated by a semi coma (;). If left in blank, all the extensions are allowed. e.g. `docx;jpg;png;xls`
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Multi-upload

This option allows to upload multiple files and store them on a table referencing a record from another table.

To use this option you first need to create a virtual field.

Multi-Upload Configuration.

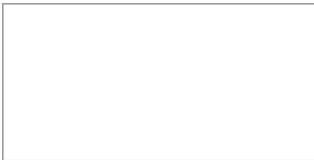
- **Amount of columns** : Amount of columns to displays the files in the Field.
- **Files sorting** : Enables files sorting after the upload.
- **Position of the delete** : Checkbox option to delete files position.
- **Position on the upload** : Position of the records when uploading.
- **Show status** : Display the upload status.
- **Table** : Select the table that will store the information of the files when uploading.

After selecting the table, you will see a list of all the fields of the chosen table, where you will associate the parameters for each field when inserting a new record or updating an existing record.

Multi-Upload fields of the table Configuration.

- **File Name** : Obtains the name of the file that is being loaded.
- **Upload** : Obtains the binary value of the file.
- **Auto Increment** : Used only when field in the database in an auto increment or similar.
- **Manual Increment** : The form will be responsible for managing the auto-increment, calculating the value of the field before inserting the record.
- **Foreign Key** : Associates the file with a value of a field in the current form.
- **Defined** : You can specify a constant value or use a global variable. `[global_var]`
- **Date of Insertion** : Obtains the data of the server when inserting the record.
- **Datetime of Insertion** : Obtains the data and time of the server when inserting the record.
- **User IP** : The field will contain the IP of the computer that is accessing the application.
- **Calculated by the Database** : The value of the field is applied in the database.
- **Calculated by the Database if empty** : The value of the field is applied in the database if not informed by the application.
- **Null** : The field will the null value.

Watch below a video about Multiupload option:



Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.

- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. Image (File Name)

Image (File Name)

General Settings

Image (File Name) field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as an Image (File Name), all the images files are stored and loaded in a directory of the server (Only the image name is stored in the database).
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Image Border** : Width of the Image border. Value in Pixels.
- **Image Height** : Image height size. Value in Pixels.
- **Image Width** : Image width size. Value in Pixels.
- **Maintain Aspect** : Maintains the original aspect ratio of the image when resizing it.
- **Progress bar** : Displays a progress bar when sending the files to the server.
- **Upload area** : Displays a drag and drop area to upload the file.
- **Increment file** : Increments the file name if the theirs an existing one with the same name in the upload folder.
- **Sub-folder** : Sub-folder's name that the files are stored. The is sub folder is relative to the directory defined for the Document upload (see the Settings). It is possible to use global variables or local variables to format the name of the sub-folder.
- **Create Sub-folder** : When enabled, it creates the sub-folder if not already created.
- **Image Caching** : Time in minutes that the image cache will be stored.
- **Hide image name** : When enabled, only the image will be displayed without it is name.
- **Files Deletion** : Deletes the files from the directory when the record is deleted from the database.
- **Display link only** : WWhen enabled, you will be presented with a link to open the image in another image with the original size of the image (without using the re-dimensioning).
- **Open in Another Window** : Allows to open the image in another window.
- **Extensions to allow** : Extensions allowed to upload, separated by a semi coma (;). If left in blank, all the extensions are allowed. e.g. docx;jpg/png;xls
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Multi-upload

This option allows to upload multiple files and store them on a table referencing a record from another table.

To use this option you first need to create a virtual field.

Multi-Upload Configuration.

- **Amount of columns** : Amount of columns to displays the files in the Field.
- **Files sorting** : Enables files sorting after the upload.
- **Position of the delete** : Checkbox option to delete files position.
- **Position on the upload** : Position of the records when uploading.
- **Show status** : Display the upload status.
- **Table** : Select the table that that will store the information of the files when uploading.

After selecting the table, you will see a list of all the fields of the chosen table, where you will associate the parameters for each field when inserting a new record or updating an existing record.

Multi-Upload fields of the table Configuration.

- **File Name** : Obtains the name of the file that is being loaded.
- **Upload** : Obtains the binary value of the file.
- **Auto Increment** : Used only when field in the database in an auto increment or similar.
- **Manual Increment** : The form will be responsible for managing the auto-increment, calculating the value of the field before inserting the record.
- **Foreign Key** : Associates the file with a value of a field in the current form.
- **Defined** : You can specify a constant value or use a global variable. [global_var]
- **Date of Insertion** : Obtains the data of the server when inserting the record.
- **Datetime of Insertion** : Obtains the data and time of the server when inserting the record.
- **User IP** : The field will contain the IP of the computer that is accessing the application.
- **Calculated by the Database** : The value of the field is applied in the database.
- **Calculated by the Database if empty** : The value of the field is applied in the database if not informed by the application.
- **Null** : The field will the null value.

Watch below a video about Multiupload option:



Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** : Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.

- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
- **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. Document (Database)

Document (Database)

General Settings

Document (Database) field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Document (Database), all the document files are stored and loaded directly from the Database.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Icon** : Displays an icon beside the field to identify the type of document.
- **Progress bar** : Displays a progress bar when sending the files to the server.
- **Upload area** : Displays a drag and drop area to upload the file.
- **File Name** : Defines the field that contains the name of the document stored in the database (the field that contains this information also needs to be stored in the database).
- **File Size** : Defines the field that contains the file size.
- **Extensions to allow** : Extensions allowed to upload, separated by a semi coma (;). If left in blank, all the extensions are allowed. e.g. `docx;jpg;png;xls`
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Multi-upload

This option allows to upload multiple files and store them on a table referencing a record from another table.

To use this option you first need to create a virtual field.

Multi-Upload Configuration.

- **Amount of columns** : Amount of columns to displays the files in the Field.
- **Files sorting** : Enables files sorting after the upload.
- **Position of the delete** : Checkbox option to delete files position.
- **Position on the upload** : Position of the records when uploading.
- **Show status** : Display the upload status.
- **Table** : Select the table that will store the information of the files when uploading.

After selecting the table, you will see a list of all the fields of the chosen table, where you will associate the parameters for each field when inserting a new record or updating an existing record.

Multi-Upload fields of the table Configuration.

- **File Name** : Obtains the name of the file that is being loaded.
- **Upload** : Obtains the binary value of the file.
- **Auto Increment** : Used only when field in the database in an auto increment or similar.
- **Manual Increment** : The form will be responsible for managing the auto-increment, calculating the value of the field before inserting the record.
- **Foreign Key** : Associates the file with a value of a field in the current form.
- **Defined** : You can specify a constant value or use a global variable. `[global_var]`
- **Date of Insertion** : Obtains the data of the server when inserting the record.
- **Datetime of Insertion** : Obtains the data and time of the server when inserting the record.
- **User IP** : The field will contain the IP of the computer that is accessing the application.
- **Calculated by the Database** : The value of the field is applied in the database.
- **Calculated by the Database if empty** : The value of the field is applied in the database if not informed by the application.
- **Null** : The field will the null value.

Watch bellow a video about Multiupload option:



Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.

- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
- **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the field.
- **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. Document (File Name)

Document (File Name)

General Settings

This option allows to upload multiple files and store them on a table referencing a record from another table.

To use this option you first need to create a virtual field.

Multi-Upload Configuration.

- **Amount of columns** : Amount of columns to displays the files in the Field.
- **Files sorting** : Enables files sorting after the upload.
- **Position of the delete** : Checkbox option to delete files position.
- **Position on the upload** : Position of the records when uploading.
- **Show status** : Display the upload status.
- **Table** : Select the table that will store the information of the files when uploading.

After selecting the table, you will see a list of all the fields of the chosen table, where you will associate the parameters for each field when inserting a new record or updating an existing record.

Multi-Upload fields of the table Configuration.

- **File Name** : Obtains the name of the file that is being loaded.
- **Upload** : Obtains the binary value of the file.
- **Auto Increment** : Used only when field in the database in an auto increment or similar.
- **Manual Increment** : The form will be responsible for managing the auto-increment, calculating the value of the field before inserting the record.
- **Foreign Key** : Associates the file with a value of a field in the current form.
- **Defined** : You can specify a constant value or use a global variable. [global_var]
- **Date of Insertion** : Obtains the data of the server when inserting the record.
- **Datetime of Insertion** : Obtains the data and time of the server when inserting the record.
- **User IP** : The field will contain the IP of the computer that is accessing the application.
- **Calculated by the Database** : The value of the field is applied in the database.
- **Calculated by the Database if empty** : The value of the field is applied in the database if not informed by the application.
- **Null** : The field will the null value.

Watch bellow a video about Multiupload option:



Multi-upload

This option allows to upload multiple files and store them on a table referencing a record from another table.

To use this option you first need to create a virtual field.

Multi-Upload Configuration.

- **Amount of columns** : Amount of columns to displays the files in the Field.
- **Files sorting** : Enables files sorting after the upload.
- **Position of the delete** : Checkbox option to delete files position.
- **Position on the upload** : Position of the records when uploading.
- **Show status** : Display the upload status.
- **Table** : Select the table that will store the information of the files when uploading.

After selecting the table, you will see a list of all the fields of the chosen table, where you will associate the parameters for each field when inserting a new record or updating an existing record.

Multi-Upload fields of the table Configuration.

- **File Name** : Obtains the name of the file that is being loaded.
- **Upload** : Obtains the binary value of the file.
- **Auto Increment** : Used only when field in the database in an auto increment or similar.
- **Manual Increment** : The form will be responsible for managing the auto-increment, calculating the value of the field before inserting the record.
- **Foreign Key** : Associates the file with a value of a field in the current form.
- **Defined** : You can specify a constant value or use a global variable. [global_var]
- **Date of Insertion** : Obtains the data of the server when inserting the record.
- **Datetime of Insertion** : Obtains the data and time of the server when inserting the record.
- **User IP** : The field will contain the IP of the computer that is accessing the application.
- **Calculated by the Database** : The value of the field is applied in the database.
- **Calculated by the Database if empty** : The value of the field is applied in the database if not informed by the application.
- **Null** : The field will the null value.

Watch bellow a video about Multiupload option:



Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Form
- 4. Barcode

Barcode

General Settings

Configuration Interface of the Barcode Field.

- **Data Type** : DataType of the field for the application.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name would be "Client Name".
- **Disabled Field** : Defines if the field will be disabled in the "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **SQL Type** : Database field type.

Values Format

Configuration Interface of the Barcode Field.

- **Type** : Type of Barcode.
- **Text** : Barcode Text for illustration purposes.
- There are **18 types of barcodes** that are listed below:

Barcode configuration interface.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. Select

Select

General Settings

Select field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Select, you can select multiple option from a combo box (Select Field).
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.
- **Use Select2** : Uses the new component for data selection, allowing searches within the select.
- **Display Select2 search area** : Sets whether to display the search field within Select2.

Initial Value Configuration Interface.

- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Field size in database** : Determines the field size in bytes. It is used to determine the max size of characters that is allowed to type in.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through Javascript or PHP.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Date of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

• Lookup Methods

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

Selecting the lookup type.

◦ Lookup Method - Automatic

Lookup used to list the values that will be displayed on the Checkbox field. These values will be recovered dynamically using the Select command.

Automatic Lookup Interface.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimitation.

- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter than (;).
- **Height** : Defines the height for the select object.
- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.
- **Link** : Allows to create a link to another form allowing to manipulate the list displayed on the select field. After the manipulation, the select object it updated automatically.

o Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

 Manual Lookup Interface.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

 Setting up Single Value Lookup

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Default** : With this option enabled, the selected item will be inserted to the table when committed.
- **Height** : Defines the height for the select object.

- **Multiple Values (delimiter)**

You can store various values for the select field. The values are separated by the informed delimiter. For example: the combination **Sport, Cinema and Tourism** selected in form will be stored like **E;C;T** in case the delimiter is ; (semi coma).

 Setting up Multiple Values (delimiter)

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** : Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Height** : Defines the height for the select object.

- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man, Single** and **Read**, in the database would be stored the following value **MSRD**.

 Setting up Multiple Values (position)

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Negative** : Value to be saved in the table in case there is not any valued selected on the field.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.
- **Height** : Defines the height for the select object.

- **Multiple Values (binary)**

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

Editing Lookup Configuration Multiple Values (Binary)

- **Label** : Text that will be displayed in the item list of the select field.
- **Height** : Defines the height for the select object.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

Displaying the original lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Lookup Method - Actual value

This lookup is used to list all the values in the selected field.

This lookup will apply a "distinct" to your SQL query.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.

- **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the field.
- **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Form
- 4. Double Select

Double Select

General Settings

Double Select field configuration Interface.

- **Data Type** : Select the type of field for the application. When set to Double Select you can have multiple options selected.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be easier to the user understand if the name is "Client Name".
- **Initial Value (type)** : Defines the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Save Variable** : Saves a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Set the name for the session variable that will receive the field value.
- **Field size in database** : Determines the size in bytes of the fields. It is used to set the max amount of characters to be typed in.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through Javascript or PHP.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Displays the data type of field in the database.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGRES and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Date of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or F can be presented like Male or Female or in a dynamic way, getting these values from the database.

Lookup Settings Display for the field. *Automatic Lookup Interface..*

- **SQL Select Statement** : Defines the SQL command that will get the values displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the `key_field` will be stored in the `table` field.

- **Height** : Set the height(lines) of the field interface.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter than (;).
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Selects another specific connection existing in the project. The select command will be done on the second connection.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**

- **Font** : Allows to choose the font type, that will be applied to the application field title.
- **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
- **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. CheckBox

CheckBox

General Settings

CheckBox field configuration Interface.

- **Data Type** : Select the type of field for the application. When set to CheckBox, you can have multiple options selected.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Initial Value (type)** : Defines the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Save Variable** : Saves a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Define the name for the session variable that will receive the field value.
- **Field size in database** : Determines the size in bytes of the fields. It is used to determine the max amount of characters to be typed in.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through JavaScript or PHP. .
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGRES and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Date of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or F can be presented like Male or Female or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

Selecting the lookup type.

- **Lookup Method - Automatic**

Lookup used to list the values that will be displayed on the CheckBox field. These values will be recovered dynamically using the Select command.

Automatic Lookup Interface.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.

- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Option check all** : Displays two options on the field to check and uncheck all.

- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- **Choose connection** : Selects another specific connection existing in the project. The select command will be done on the second connection.

o **Lookup Method - Manual**

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

Manual Lookup Interface.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).

- **Single Value** :

Setting up Single Value Lookup

- **Label** : Text that will be displayed in the item list of the checkBox.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Default** : With this option enabled, the selected item will be inserted to the table when committed.
- **Columns** : Set the amount of columns, for the list of items.

- **Multiple Values (delimiter)**

You can store various values for the checkBox field. The values are separated by the informed delimiter. For example: the combination **Sport, Cinema and Tourism** selected in form will be stored like **E;C;T** in case the delimiter is ; (semi coma).

Setting up Multiple Values (delimiter)

- **Label** : Text that will be displayed in the item list of the checkBox.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Columns** : Set amount of columns, for the list of items.

- **Multiple Values (position)**

Stores a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man, Single and Read**, in the database would be stored the following value **MSRD**.

Setting up Multiple Values (position)

- **Label** : Text that will be displayed in the item list of the checkBox.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Negative** : Value to be saved in the table in case there is not any valued selected on the field.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.
- **Columns** : Set the amount of columns, for the list of items.

- **Multiple Values (binary)**

Stores a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports, Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute	Value	Lookup	Description
-----------	-------	--------	-------------

Attribute Value	Lookup Description
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

Setting up Multiple Values (Binary)

- **Label** : Text that will be displayed in the item list of the checkBox.
- **Columns** : Allows you to inform the amount of columns, for the list of items.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Saves all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Refreshes the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Option check all** : Displays two options on the field to check and uncheck all.

- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** : Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** : Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** : Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. Radio

Radio

General Settings

Radio field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Radio, your allowed to select one of the options listed.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Field size in database** : Determines the size in bytes of the fields. It is used to determine the max amount of characters to be typed in.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through Javascript or PHP. .
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGRES and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Date of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

Selecting the lookup type.

- **Lookup Method - Automatic**

Lookup used to list the values that will be displayed on the Checkbox field. These values will be recovered dynamically using the Select command.

Automatic Lookup Interface.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.

- **Columns** : Allows you to inform the amount of columns, for the list of items.

- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
 - **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

o Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

Manual Lookup Interface.

- **Label** : Text that will be displayed in the item list of the radio.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Default** : With this option enabled, the selected item will be inserted to the table when committed.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

Displaying the original lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Ajax Settings

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - o **Font** : Allows to choose the font type, that will be applied to the application field title.
 - o **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - o **Font Color** :Allows to choose a color for the font from the color pallet.
 - o **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - o **Bold** : Applies the bold style to the font.
 - o **Underline** : Applies the underline style to the font.
 - o **Border Size** : Applies the border size to the title of the field.
 - o **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - o **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - o **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - o **Width** : To define a width for the title of the field.
 - o **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - o **Font** : Allows to choose the font type, that will be applied to the application field.
 - o **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - o **Font Color** :Allows to choose a color for the font from the color pallet.
 - o **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - o **Bold** : Applies the bold style to the font.
 - o **Underline** : Applies the underline style to the font.
 - o **Border Size** : Applies the border size to the field.
 - o **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - o **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - o **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - o **Width** : To define a width for the field.
 - o **Height** : To define a height for the field.
- **CSS of the Input Object**
 - o **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - o **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - o **Font Color** :Allows to choose a color for the font from the color pallet.
 - o **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - o **Bold** : Applies the bold style to the font.
 - o **Underline** : Applies the underline style to the font.
 - o **Border Size** : Applies the border size to the Input Object.
 - o **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - o **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - o **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - o **Width** : To define a width for the Input Object.
 - o **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Form
- 4. Text Auto-Complete

Text Auto-Complete

General Settings

Text auto complete field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Text auto complete, you can inform a value based on the select statement of the lookup settings and it will manipulate an internal Text for the data.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You'll inform the Defined Value here.
- **Use Select2**: Uses the new component for data selection, allowing searches within the select.
- **Amount of characters for the Select2**: Sets the number of characters to start the search in Select2.
- **Amount of lines for the Select2**: Sets the maximum number of rows to list the search result in Select2.
- **Width for the Select2**: Sets a width for the area for the Select2.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Complete to the Left** : Allows to define the character that will be used to complete the value to the left that the user typed in to the max size of the field defined in the Field size in database option.
- **Field size in database** : Determines the field size in bytes. it is used to determine the max size of characters that is allowed to type in.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Validate on submit** : Validate the field only when the form is submitted.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface

- **Case Settings** : Allows to convert the letters of the field when losing the focus. The options are:
 - **Upper Case** : All in Upper Case.
 - **Lower Case** : All in Lower Case.
 - **Capitalize first word** : Capitalizes the first letter of the word.
 - **Capitalize all words** : Capitalizes the first letter of all the words.
- **Minimum Size** : Allows to define the minimum amount of characters accepted for the field.
- **Maximum Size** : Allows to define the maximum amount of characters accepted for the field.
- **Allowed Characters** : Allows to select a set of characters that can be typed into the field. The options are:
 - **All** : Allows any type of character to be typed into the field
 - **Selected** : Defines a set of characters allowed. Using the configuration below, will be allowed to type letters, numbers and other characters (defined in the attribute More). The letters a, b and c (defined in the Less Attribute) are not allowed.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGRES and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is being accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field.
- **Rows** : Allows to define the amount of records displayed in the field.
- **Width** : Defines the width the size of capture box (Capture Text).
- **Search options** : Allows to define the search settings of the field(Start equal to, Any part and End equals to).
- **Capture Text** : When not enabled, you will only view the field to inform the data. See an example below. lookup Settings configuration Interface.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup. lookup Settings validation configuration Interface.
- **Show label with the description** : Displays a label with the description.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

-
- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
 - **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
 - **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS

Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Form
4. Number Auto-Complete

Number Auto-Complete

General Settings

Number auto complete field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Number auto complete, you can inform a value based on the select statement of the lookup settings and it will manipulate an internal number for the data.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Complete to the Left** : Allows to define the character that will be used to complete the value to the left that the user typed in to the max size of the field defined in the Field size in database option.
- **Field size in database** : Determines the field size in bytes. It is used to determine the max size of characters that is allowed to type in.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Validate on submit** : Validate the field only when the form is submitted.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface

- **Case Settings** : Allows to convert the letters of the field when losing the focus. The options are:
 - **Upper Case** : All in Upper Case.
 - **Lower Case** : All in Lower Case.
 - **Capitalize first word** : Capitalizes the first letter of the word.
 - **Capitalize all words** : Capitalizes the first letter of all the words.
- **Minimum Size** : Allows to define the minimum amount of characters accepted for the field.
- **Maximum Size** : Allows to define the maximum amount of characters accepted for the field.
- **Allowed Characters** : Allows to select a set of characters that can be typed into the field. The options are:
 - **All** : Allows any type of character to be typed into the field
 - **Selected** : Defines a set of characters allowed. Using the configuration below, will be allowed to type letters, numbers and other characters (defined in the attribute More). The letters a, b and c (defined in the Less Attribute) are not allowed.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field.

- **Rows** : Allows to define the amount of records displayed in the field.
- **Width** : Defines the width the size of capture box (Capture Text).
- **Search options** : Allows to define the search settings of the field(Start equal to, Any part and End equals to).
- **Capture Text** : When not enabled, you will only view the field to inform the data. See an example below. *lookup Settings configuration Interface.*
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.
 lookup Settings validation configuration Interface.
- **Show label with the description** : Displays a label with the description.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Form
- 4. Signature

Signature

General Settings

Configuration Interface of the Signature Field.

The signature field will help you creating more sophisticated forms and making it possible to store signatures in your database. Inside our development environment we have specific settings that will help you to customize your field, those options are:

- **Data Type** : You can define the type of field for the application. When it is defined as a text, it accepts letters, numbers and special characters.
- **Label** : Lets you define a label to the field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Background color** : Defines a color to the field background by using a valid hexadecimal color value or by choosing a color from the color palette.
- **Pen color** : Set a color to the pen by using a valid hexadecimal color value or by choosing a color from the color palette.
- **Width** : Lets you define a width to the field.
- **Height** : Set a height to the field.
- **Subtitle** : Defines the subtitle that will be displayed beside the field.
- **Initial Value** : Lets you define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
- **Save Variable** : Save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Set the name for the session variable that will receive the field value.
- **Disabled Field** : Define if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : Displays the HTML Object that will be used to display the field in the form.
- **SQL Type** : Displays the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of ScriptCase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **CSS of the Title**
 - **Font** : Select the font type, that will be applied to the application field title.
 - **Font Size** : Defines the font size, that will be applied to the application field title.
 - **Font Color** : Choose a color to the font by using a valid hexadecimal color value or from the color picker.
 - **Background Color** : You can define the color for the field by using a valid hexadecimal color value or from the color picker.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Set the underline style to the font.
 - **Border style** : Choose a style for the border.
 - **Collapse** : Defines the collapse for the border.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Choose a color for the border, using a color palette to apply to the title.
 - **Horizontal Alignment** : Position the label of the field in the wanted position (left,right,center and justify).
 - **Vertical Alignment** : Position the label of the field in the wanted position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : Defines a width for the title of the field.
 - **Height** : Set a height for the title of the field.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

□

1. [Home](#)
2. [Applications](#)
3. Form
4. Rating

Rating

General Settings

This field allows the developer to create a field for rating using stars (or any other image), where the final user will be able to select a rate option according to the configuration made by the developer.

Description of the attributes available for configuration.

Data type

Define the type of the field in the application. In this case we must select the type **Rating**.

Label

Define the application title that will be showed when the application is executed. The terminology used in the interface of your application is very important for the usability, and we need to use familiar terms for the final user, instead of use terms from the system.

Example, this field "rating" that is named **cmp_rating** however, the customer would have a better understanding if the label uses **Product Rating**.

Besides use a fixed text, the **Label** attribute allows the use of **Langs** to define the field title, allowing the [internationalization of your application](#).

Subtitle

Define the subtitle that will be shown in the field below the ratings. **Example**: "Thank you for your feedback!".

As in the **Label**, the **subtitle** attribute also allows the use of **Langs** for [internationalization of your application](#).

Amount of icons

Define the number of icons that will be displayed in the field. The value set in this attribute must be according to the way the evaluation will be realized.

To evaluate using Stars normally it is used five stars, so , we must to inform 5 in this attribute. After we set the images that will be displayed in the attribute **Enabled icon** and **Disabled icon** the result will be the same as in the example:

To use Like/Dislike **for example** we must inform 1 in the **amount of icons**, so only one image will be displayed and will change according to the click. In this case, the storage in the database will be 1 for like (enabled icon) and 0 for dislike (disabled icon). **You can see the field configured below as like/dislike:**

Enabled icon

Define the icon that will be shown for the amount selected. Following the like/dislike example, in this option, we must insert the Like image . We must do the same for a star evaluation, where for default we must insert the stars filled .

Disabled icon

Define the icon that will be shown for the not selected icons. In this attribute, we set an image to represent the opposite of the previous images set in the **Enabled icon**.

Still referring the example of the Like/Dislike, in this attribute we must set the Dislike .

As in the stars example, we must insert the image with the empty stars .

Initial Value (type)

Allow the definition of an initial value for the field when the form application is in insert mode. The only option available in this field is **Defined**. Selecting this option the attribute **Initial Value** will be showed for set the value.

Example: The initial value is 3 when inserting a new record, the rate field will initiate with 3 stars already selected.

The value defined in this attribute will overlap any value defined previously.

Save variable

Allow saving a session variable ([Global variable](#)) with the field value for being used in other applications.

Example: In the login form, the user name can be saved in the session and used to be displayed in the header of other applications.

Variable Name

In this attribute, we must define a name for the **session variable** that was enabled previously to receive the field value.

We must inform only the variable name, - **var_rating**.

We can recover the value as a [global variable](#).

HTML Type

HTML object used to show the form field.

SQL Type

Inform the type of the field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of ScriptCase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **CSS of the Title**

- **Font** : Select the font type, that will be applied to the application field title.
- **Font Size** : Defines the font size, that will be applied to the application field title.
- **Font Color** : Choose a color to the font by using a valid hexadecimal color value or from the color picker.
- **Background Color** : You can define the color for the field by using a valid hexadecimal color value or from the color picker.
- **Bold** : Applies the bold style to the font.
- **Underline** : Set the underline style to the font.
- **Border style** : Choose a style for the border.
- **Collapse** : Defines the collapse for the border.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Choose a color for the border, using a color palette to apply to the title.
- **Horizontal Alignment** : Position the label of the field in the wanted position (left,right,center and justify).
- **Vertical Alignment** : Position the label of the field in the wanted position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : Defines a width for the title of the field.
- **Height** : Set a height for the title of the field.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

Pages

A page is a container of blocks. Each application has at least one page by default. In Form, Control, and Search applications, there is possible to create many pages. The pages are like Tabs, where each tab contains one or more blocks.

Pages (available only in the Form, Control and Search applications) configuration Interface.

See the example below of the Form application using two Pages: General Data and Documents.

Form Application using Pages feature.

Pages Settings

The form application already comes with a default page, identified as "Pag1", but it is possible to rename it. Use the pages when you have an application that contains many fields. A form with more than 20 fields in a vertical way is challenging to use. So you could arrange the fields into the blocks and the blocks into the pages.

PAGES SETTINGS														
ATTRIBUTE	VALUE	DESCRIPTION												
<div style="border: 1px solid #ccc; padding: 5px; margin: 5px auto; width: 80%;"> <p style="text-align: center; margin: 0;">Edit the Page Settings</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Op</th> <th>Name</th> <th>Title</th> <th>Icon</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">🗑</td> <td>Pag1</td> <td>Pag1</td> <td style="text-align: center;"></td> </tr> <tr> <td style="text-align: center;">+</td> <td>Pag2</td> <td>pag2</td> <td style="text-align: center;"></td> </tr> </tbody> </table> <p style="text-align: center; margin: 5px 0;">Include</p> </div>			Op	Name	Title	Icon	🗑	Pag1	Pag1		+	Pag2	pag2	
Op	Name	Title	Icon											
🗑	Pag1	Pag1												
+	Pag2	pag2												
Minimum tab width	<input type="text"/>													
Maximum tab width	<input type="text"/>													
Font	<input type="text"/> Aa													
Font Size	<input type="text"/>													
Selected Font Color	<input type="text"/>													
Selected Background Color	<input type="text"/>													
Non Selected Font Color	<input type="text"/>													
Non Selected Background Color	<input type="text"/>													

Pages Configuration Interface.

Including A New Page,

To include a new page, enter the information about the name and the label of the new page, select an image if you wish, then click on the button "Include".

Deleting A Page

To delete a page, click on the Trash icon corresponding to the page line.

Common Settings

Font

Set the font-family of the pages titles. By clicking on the right side icon, you can select the font-family from a list.

Font Size

Set the font size of the pages titles.

Selected Font Color

Set the font color of the selected page.

Selected Background Color

Set the Background Color of the selected page.

Non Selected Font Color

Set the font color of the non selected pages.

Non Selected Background Color

Set the Background Color of the non selected pages.

Blocks

Blocks are "containers" where you can position the application fields of Forms, Controls, or Slide Grids.

Scriptcase creates applications with one block by default. You can add more blocks as you wish, to organize it in the best way.

See below, the Columns Organization, and where you can define the position of the next block: beside or below the current one.

		Block	Title	Label	Fields		Organization			
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
+	form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▾	1	Beside ▾	Below ▾	100%	Start open ▾
Pag2										
+	Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▾	1	Beside ▾	Below ▾	100%	Start open ▾

Application Block configuration

On the left side of each block, there are two icons, the first one to edit the information of the block and the second one to delete the block.

Organizing The Position Of The Blocks

See below how to modify the display order of the Blocks in one Page.

Click and drag the block that you desire to modify to its new position.

Block		Title		Label		Fields			Organization	
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
		form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Pag2										
		Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open

Application Block Display configuration

- see how to remove a block from the display

Click on the block desired and drag it to the item "Blocks not Shown". This way, you can also drag the block to another page if desired. See the images below.

Block		Title		Label		Fields			Organization	
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
		form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Pag2										
		Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Blocks not shown										
		Block 3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open

Application Block Display configuration

Block		Title		Label		Fields			Organization	
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
		form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Pag2										
		Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Blocks not shown										
		Block 3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open

Application Block Display configuration

Block

- **ame:** The name that identifier the Block.
- **abel:** Title of the block to display in the application.

Title

- **isplay:** It controls the display of the block title.

Label

- **isplay:** It controls the display of the field labels of the block.
- **osition:** Options to display label :
 1. **bove:** Display the label above the field.
 2. **eside:** Display the label beside the field.
 3. **elow:** Display the label below the field.

Fields

- **olumns:** Amount of columns side by side in the block.
- **osition:** The way to display the fields in the block:
 1. **elow:** Display the fields one below the other respecting the number of columns.
 2. **eside:** Display the fields one beside the other respecting the number of columns.
 3. **ine:** Display the fields one beside the other with no tabulation.

Organization

- **ext:** The way to display the blocks in the page:
 1. **elow:** Set to show the following block below the current one.
 2. **eside:** Set to show the following block beside the current one.
 3. **abs:** Set to show the following block in a different tab then the current one.
- **idth:** Set the block width in pixels or percentages. Use the symbol "%" to indicates the value in percentage.
- **ollapse:** Enables the option to close the block.

Create A New Block

To include new blocks in an Application, click on the button . Then, enter the name and label of the block in the following interface and finish by click on Create.

Add New Block

NAME

LABEL

Creating application blocks configuration


Name

Name of the Block.

Label

Title of the block to display in the application.

Edit Blocks

To edit a block, click on the icon , that is on the left side of the block. Then you can see the following interface to define the parameters of the blocks. Click on Save to finish.

ATTRIBUTE	VALUE
Name	<input type="text" value="form_orders"/>
Title	<input type="text" value="form_orders"/>
Display Title	<input type="radio"/> Yes <input checked="" type="radio"/> No
Title Font	<input type="text" value="Aa"/>
Font Size	<input type="text" value=""/>
Font Color	<input type="text" value=""/>
Background Color	<input type="text" value=""/>
Background image	<input type="text" value=""/>
Title Height	<input type="text" value="20"/> pixels
Horizontal Alignment	<input type="text" value=""/>
Vertical Alignment	<input type="text" value=""/>
Display Label	<input checked="" type="radio"/> Yes <input type="radio"/> No
Columns	<input type="text" value="1"/>
Columns Width	<input type="text" value="Calculated"/>
Label Color	<input type="text" value=""/>
Fields Organization	<input type="text" value="Beside"/>
Label Position	<input type="text" value="Beside"/>
Next Block	<input type="text" value="Below"/>
Border Color	<input type="text" value=""/>
Border Width	<input type="text" value="0"/> pixels
Block Width	<input type="text" value="100%"/>
Block Height	<input type="text" value=""/>
Cell Spacing	<input type="text" value=""/> pixels
Collapse	<input type="text" value="Start open"/>

Application Block editing interface

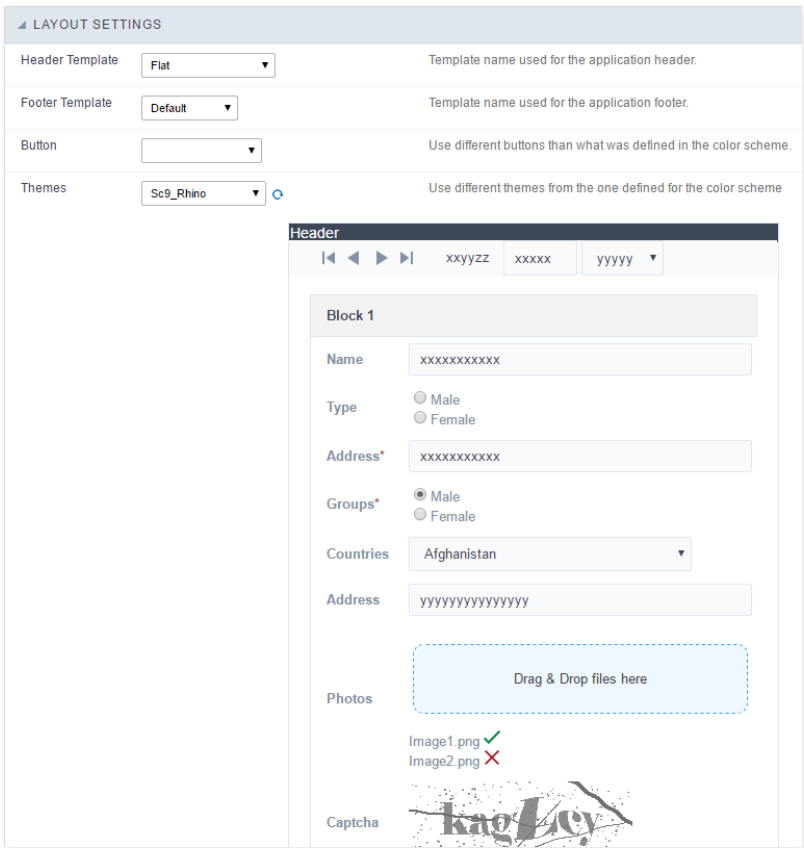
Name

Name of the block. ##### Title
 Block title for display. ##### Display Title
 This option, when active, allows displaying the block title. ##### Title Font
 Set the font family of the block title. ##### Font Size
 Set the font size of the block title. ##### Font Color
 Set the font color of the block title. ##### Background Color
 Set the Background Color of the block title. ##### Background image
 Set a Background image for the block title. ##### Title Height
 Height in pixels of the block title line. ##### Horizontal Alignment
 Horizontal Alignment of the block title (Left, Center, and Right). ##### Vertical Alignment
 Vertical Alignment of the block title (Top, Middle, and Bottom). ##### Display Label
 Display the labels of the fields in the block. ##### Columns
 Amount of field columns in a block. ##### Columns Width
 Set the field column width of the block. ##### Label Color
 Color of the field labels. ##### Fields Organization
 The way to display the fields in the block. ##### Label Position
 Set the position of the field labels of the block. ##### Next Block
 Set the position of the following block relating to the current one. ##### Border Color
 The border Color for the block. ##### Border Width
 The border Width for the block. ##### Block Width
 The width for the block. ##### Block Height
 The Height for the block. ##### Cell Spacing
 The Cell Spacing in the block. ##### Collapse
 It enables the option to close the block.

This module lets you editing display options for the application. You can apply visualization themes, organize blocks, define values, and the display format of the Header and Footer.

Layout Settings

ScriptCase creates an app by using default values defined per project, but you can change those values for a specific application on this interface.



Application Layout Settings

Header Template

Allows choosing the look of the application Header.

Footer Template

Allows choosing the look of the application Footer.

Button

Allows choosing the button theme for the application.

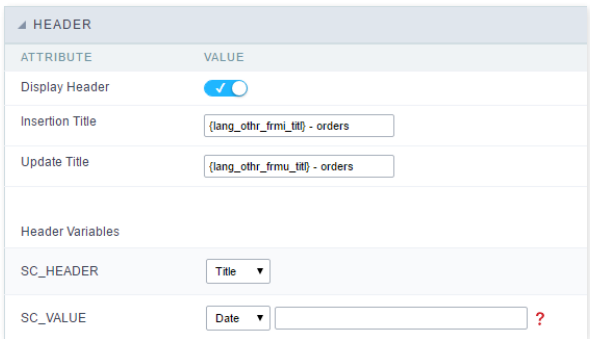
Themes

Choose one of the selected themes in the project properties. It defines the look of the application, like colors, fonts, and others.

Header & Footer

Header

Here we can define if we want to display the header or not, the application title, and fill the header variables. The header looks according to the selected template in the Layout settings interface.



Application Header Configuration

Display Header


This option determines if we want to display the header or not.

Title

Allows informing the title of the application.

Header Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **itle**: It displays the value of "**Application Title**" in the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Depending on the Application, you may have more than one title option.

Footer

FOOTER	
ATTRIBUTE	VALUE
Display Footer	<input type="checkbox"/>
Footer Variables	
NM_FOOTER_DATA	<input type="text"/>
NM_FOOTER_LOGO	<input type="text"/>
NM_FOOTER_TITLE	<input type="text"/>

Application Footer Settings


This page may change depending on the footer template chosen in the Layout Settings.

Display Footer

This option determines if we want to display the footer or not.

Footer Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Within the events you can include Scriptcase global, local variables, libraries and macros, PHP, JavaScript, HTML and CSS codes.

onApplicationInit

This event occurs only once, when the application is loaded and before the application executes the SQL. It can be used for data processing or checking variables.

Example: `if ([glo_var_dept] != 'financial'){ sc_redir(app_x.php); }`

onNavigate

This event occurs when navigating between the application pages.

OnScriptInit

This event occurs when the application scripts are started. It runs before the OnLoad events(when present), however, it will always be executed when running, reloading or navigating the application.

onLoadRecord

This event is executed at the moment the record line is loading. The OnRecord event is a loop that loads the records by each row.

onLoad

This event runs when the application is finished loading.

onRefresh

This event runs when the refresh option of the application is enabled.

onValidate

This event runs when validating the information of the fields, when submitting the form.

onValidateFailure

This event runs when the form validation gives an error.

onValidateSuccess

This event runs when the form validation does not give an error.

onBeforeInsert

This event is executed before inserting a record from the application.

onAfterInsert

This event is executed after inserting a record from the application.

onBeforeInsertAll

This event runs before inserting all the selected records from the Multiple Records, Editable Grid and Editable Grid (View) Form.

onAfterInsertAll

This event runs after inserting all the selected records from the Multiple Records, Editable Grid and Editable Grid (View) Form.

onBeforeUpdate

This event is executed before updating a record from the application.

onAfterUpdate

This event is executed after updating a record from the application.

onBeforeUpdateAll

This event runs before updating all the selected records from the Multiple Records, Editable Grid and Editable Grid (View) Form.

onAfterUpdateAll

This event runs after updating all the selected records from the Multiple Records, Editable Grid and Editable Grid (View) Form.

onBeforeDelete

This event runs before deleting the records from the Form.

onAfterDelete

This event runs after deleting the records from the Form.

onBeforeDeleteAll

This event runs before deleting all the selected records from the Multiple Records, Editable Grid and Editable Grid (View) Form.

onAfterDeleteAll

This event runs after deleting all the selected records from the Multiple Records, Editable Grid and Editable Grid (View) Form.

OnClick

The ajax event OnClick is executed when the field that it's based on is clicked.

- reating a new ajax event



Creating a new ajax event

- electing a field

Creating Ajax Events	
Select the field to create an event	contacttitle
Select event for Ajax processing	onClick
Fields that will be passed as parameters Double click to check or uncheck	<ul style="list-style-type: none"> customerid companyname contactname contacttitle birthdate country regionid stateid city address postalcode phone
<input type="button" value="Create Event"/>	

Selecting a field of the ajax event

Choose a field to create an event To define in which field the event will be add to.

Choose an event that run the ajax Defines which event will be added to the field.

OnChange

The ajax event OnChange is executed when the value of the field that it's based on is modified.

- reating a new ajax event



Creating a new ajax event

- electing a field

Creating Ajax Events	
Select the field to create an event	contacttitle
Select event for Ajax processing	onClick
Fields that will be passed as parameters Double click to check or uncheck	<ul style="list-style-type: none"> customerid companyname contactname contacttitle birthdate country regionid stateid city address postalcode phone
<input type="button" value="Create Event"/>	

Selecting a field of the ajax event

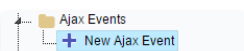
Choose a field to create an event To define in which field the event will be add to.

Choose an event that run the ajax Defines which event will be added to the field.

OnBlur

The ajax event OnBlur is executed when the focus is removed from the field that event is based on.

- reating a new ajax event



Creating a new ajax event

- electing a field

Creating Ajax Events

Select the field to create an event

Select event for Ajax processing

Fields that will be passed as parameters
Double click to check or uncheck

- customerid
- companyname
- contactname
- contacttitle
- birthdate
- country
- regionid
- stateid
- city
- address
- postalcode
- phone

Create Event

Selecting a field of the ajax event

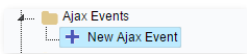
Choose a field to create an event To define in which field the event will be add to.

Choose an event that run the ajax Defines which event will be added to the field.

OnFocus

The ajax event OnFocus is executed when the field that it's based on is applied with a focus.

- reating a new ajax event



Creating a new ajax event

- electing a field

Creating Ajax Events

Select the field to create an event

Select event for Ajax processing

Fields that will be passed as parameters
Double click to check or uncheck

- customerid
- companyname
- contactname
- contacttitle
- birthdate
- country
- regionid
- stateid
- city
- address
- postalcode
- phone

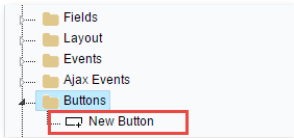
Create Event

Selecting a field of the ajax event

Choose a field to create an event To define in which field the event will be add to.

Choose an event that run the ajax Defines which event will be added to the field.

In addition to the applications buttons, you can also create new manual buttons. These buttons will be placed on the application toolbar.

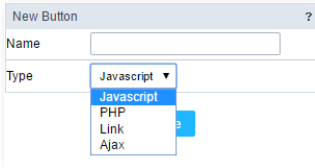


Creating new buttons

Creating a button

To create a new button, click the "new button" and type a name and the button type.

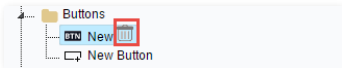
Form button types: (JavaScript, PHP, and Ajax).



Form button types

Deleting a button

To delete a button click on the icon next to the name of the button in the application menu (recycle bin).



Deleting a button

JavaScript

Display Mode

You can configure the display mode of the javascript button in Image, Button or Link.

Button

Button Settings: JavaScript	
ATTRIBUTE	VALUE
Display Mode	Button ▼
Label	JavaScript
Hint	
Confirmation Message	
Type	JavaScript

Setting up Javascript Button.

- Display Mode** You can select the display mode for the javascript button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Type** Description of the created button.

Image

Button Settings: JavaScript	
ATTRIBUTE	VALUE
Display Mode	Image ▼
Icon	<input type="text"/>
Hint	
Confirmation Message	
Type	JavaScript

Setting up Javascript Button.

- Display Mode** You can select the display mode for the javascript button in this option.
- Icon** Allows you to inform the icon that will be displayed on the button while the execution of the application.

- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Type** Description of the created button.

Link

► Button Settings: JavaScript

ATTRIBUTE	VALUE
Display Mode	Link ▾
Label	JavaScript
Hint	
Confirmation Message	
CSS Style	default ▾
Type	JavaScript

Setting up Javascript Button.

- Display Mode** You can select the display mode for the ajax button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- CSS Style** Name of the CSS class, style created in the layout editor.
- Type** Description of the created button.

Code Block

► JavaScript Code

Theme default ▾

```
1
```

Javascript button coding block.

In this block, only JavaScript is accepted.

PHP

Display Mode

You can configure the display mode of the PHP button in Image, Button or Link.

Button

► Button Settings: PHP

ATTRIBUTE	VALUE
Display Mode	Button ▾
Label	PHP
Hint	
Confirmation Message	
Reload quantity of records	No ▾
Type	PHP
Target	Same Window ▾

Setting up PHP Button.

- Display Mode** You can select the display mode for the link button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Reload quantity of records** This option is used to update the amount of records in the application.
- Type** Description of the created button.
- Target** Defines the window destination where the code will run (Open in the same Window, Open in another Window, Modal).

Image

ATTRIBUTE	VALUE
Display Mode	Image
Icon	<input type="text"/>
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
Reload quantity of records	No
Type	PHP
Target	Same Window

Setting up PHP Button.

- Display Mode** You can select the display mode for the link button in this option.
- Icon** Allows you to inform the icon that will be displayed on the button while the execution of the application.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Reload quantity of records** This option is used to update the amount of records in the application.
- Type** Description of the created button.
- Target** Defines the window destination where the code will run (Open in the same Window, Open in another Window, Modal).

Link

ATTRIBUTE	VALUE
Display Mode	Link
Label	PHP
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
CSS Style	default
Reload quantity of records	No
Type	PHP
Target	Same Window

Setting up PHP Button.

- Display Mode** You can select the display mode for the link button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- CSS Style** Name of the CSS class, style created in the layout editor.
- Reload quantity of records** This option is used to update the amount of records in the application.
- Type** Description of the created button.
- Target** Defines the window destination where the code will run (Open in the same Window, Open in another Window, Modal).

Code Block

ATTRIBUTE	VALUE
Code	*Ajax button coding block. *

In this block, you can use macros, PHP code and JavaScript.

Link

Display Mode

You can configure the display mode of the link button in Image, Button or Link.

Button

ATTRIBUTE	VALUE
Display Mode	Button
Label	Link
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
Type	Link


[Link](#)

Setting up Link Button.

Display Mode	You can select the display mode for the link button in this option.
Label	Text of the button that will be display in the application while executing.
Hint	Hint message for the button. (Displayed when the mouse hovers the button).
Confirmation Message	Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
Type	Description of the created button.

Image

► Button Settings: Link

ATTRIBUTE	VALUE
Display Mode	Image ▾
Icon	<input type="text"/> 
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
Type	Link

[Link](#)

Setting up Link Button.

Display Mode	You can select the display mode for the link button in this option.
Icon	Allows you to inform the icon that will be displayed on the button while the execution of the application.
Hint	Hint message for the button. (Displayed when the mouse hovers the button).
Confirmation Message	Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
Type	Description of the created button.

Link

► Button Settings: Link

ATTRIBUTE	VALUE
Display Mode	Link ▾
Label	Link <input type="text"/>
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
CSS Style	default ▾
Type	Link

[Link](#)

Setting up link Button.

Display Mode	You can select the display mode for the ajax button in this option.
Label	Text of the button that will be display in the application while executing.
Hint	Hint message for the button. (Displayed when the mouse hovers the button).
Confirmation Message	Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
CSS Style	Name of the CSS class, style created in the layout editor.
Type	Description of the created button.

Setting Up The Link

- electing the applications

Link between applications - Application List

Select the application that will be called:

Application All By folder By type

- calendar_events
- chart_customers
- dashboard
- form_customers
- form_employees
- form_orders
- form_sec_users
- grid_categories

[◀ Back](#) [Next ▶](#) [Help](#)

Choosing the application for the button link.

You should select an application to be called from the button link.

- ink Parameters

Choosing the parameters for the button link.

Field Allows you to use an existing field from the current application as a parameter for the link.

Variable Allows you to use a global variable from the current application as a parameter for the link.

Fixed Allows you to inform a fixed value as a parameter for the link.

Empty No value will be passed as a parameter for the link.

- ink Properties (Grid)

Configuring the properties for the link button when the destined application is a Grid.

Link Operation Mode How the link will open (Open in the same Window, Open in another Window, Modal).

Exit URL for the target application URL or an application that be redirect to when exiting the Grid application.

Initial Mode Allows you to define the initial mode of the grid application (Search or Grid).

Number of Lines Allows you to define the amount of lines displayed in the Grid.

Number of Columns Allows you to define the amount of columns displayed in the Grid.

Paging Enable the paging in the Grid.

Display Header Enable the Grid Header.

Active Navigation Buttons Enable the navigation button (First, Back, Next and Last) in the Grid.

- ink Properties (Form)

Configuring the properties for the link button when the destined application is a Form.

Link Operation Mode How the link will open (Open in the same Window, Open in another Window, Modal).

Exit URL for the target application URL or an application that be redirect to when exiting the Form application.

Enable insert button on target application Enable the "New" button in the Form Application.

Enable update button on target application Enable the "Update" button in the Form Application.

Enable delete button on target application Enable the "Delete" button in the Form Application.

Enable navigation button on target application Enable the navigation button (First, Back, Next and Last) in the Form.

Enable button to edit a grid record Enable the buttons that allow you to edit the records of a Grid

Ajax

Display Mode

You can configure the display mode of the Ajax button in Image, Button or Link.


Button

▶ Button Settings: Ajax	
ATTRIBUTE	VALUE
Display Mode	Button ▾
Label	Ajax
Hint	
Confirmation Message	
Type	Ajax
Show in Update mode	<input checked="" type="radio"/> Yes <input type="radio"/> No
Show in the inserting mode	<input type="radio"/> Yes <input checked="" type="radio"/> No

Setting up Ajax Button.

- Display Mode** You can select the display mode for the Ajax button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Type** Description of the created button.

Image

▶ Button Settings: Ajax	
ATTRIBUTE	VALUE
Display Mode	Image ▾
Icon	<input type="text"/> 
Hint	
Confirmation Message	
Type	Ajax
Show in Update mode	<input checked="" type="radio"/> Yes <input type="radio"/> No
Show in the inserting mode	<input type="radio"/> Yes <input checked="" type="radio"/> No

Setting up Ajax Button.

- Display Mode** You can select the display mode for the Ajax button in this option.
- Icon** Allows you to inform the icon that will be displayed on the button while the execution of the application.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Type** Description of the created button.

Link

▶ Button Settings: Ajax	
ATTRIBUTE	VALUE
Display Mode	Link ▾
Label	Ajax
Hint	
Confirmation Message	
CSS Style	default ▾
Type	Ajax
Show in Update mode	<input checked="" type="radio"/> Yes <input type="radio"/> No
Show in the inserting mode	<input type="radio"/> Yes <input checked="" type="radio"/> No

Setting up Ajax Button.

- Display Mode** You can select the display mode for the ajax button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- CSS Style** Name of the CSS class, style created in the layout editor.
- Type** Description of the created button.

Code Block

▶ PHP Code	
1	<input type="text"/>

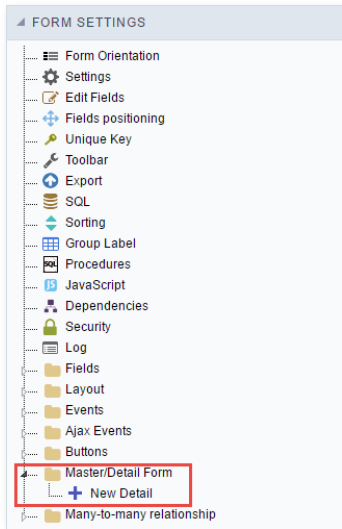
Ajax button coding block.

In this block, you can use macros, Ajax code and JavaScript.

This option allows you to create forms to update, on a single screen, tables with 1:N relationship.

New detail

In the application menu, you can create a new connection by clicking on the "New detail" option from Master/detail folder.



Creating a new detail.

Setting Name And Label

Interface to set link name and label. It is going to be displayed within the master application as a field. Name can not contain spaces nether special characters.

 A form titled 'Include Master/Detail Form'. It has two input fields: 'NAME' and 'LABEL'. Below the fields are two buttons: 'Create' and 'Help'.

Interface to set link name and label.

Application List

Only the form types: Editable Grid or Editable Grid View can be used as an application detail. You must create the form detail application before you create the Master/Detail link, because you will need to select the application during the link creation.

 A screenshot of the 'Link between applications - Application List' dialog. It features a search bar with the text 'Select the application that will be called:'. Below the search bar are three radio buttons: 'All', 'By folder', and 'By type'. A list of applications is shown with radio buttons next to each: form_employees, form_orders (selected), grid_categories, grid_customers, grid_events, and grid_orders. At the bottom are three buttons: '< Back', 'Next >', and 'Help'.

Interface to configure the detail application.

Parameters Definition

In the left side column will be displayed all master form fields, and on the right side column are listed the detail form fields. You will need to associate the master form primary keys with the detail form foreign keys. The 1:N relationship between the tables must to exist also within the database.

Link between applications -

select the foreign key fields

FIELD	KEY
customerid	customerid ▼
companyname	▼
contactname	▼
contacttitle	▼
birthdate	▼
country	▼
regionid	▼
stateid	▼
city	▼
address	▼
postalcode	▼
phone	▼
fax	▼
cityid	▼
creditlimit	▼
cardtype	▼
cardnumber	▼
Files	▼

« Back Save

Interface to configure the link parameters.

Link properties

After the link creation, you can change the properties by clicking on the link and selecting the properties option.

Form properties

Enable insert button on target application

Enable update button on target application

Enable delete button on target application

Enable navigation button on target application

Maintain the WHERE clause in the target application

Iframe properties

Iframe width

Iframe height

Other properties

Paging ▼

Lines per page

when deleting ▼

Edit link Save

Link properties.

- **Attributes**
 - **Form properties** : Within this option you can select the buttons that are going to be displayed in the detail form.
 - **Iframe properties** : This option allows you to configure the iframe height and width (in pixels) where the detail form is going to be displayed.
 - **Other properties**
 - **Editable Grid View** : Allows you to select if the detail form will be an Editable Grid view or not.
 - **Paging** : Allows you to select your form detail paging. The options are:
 - **Partial** : This option will set page layout as partial, with a limited records per page, according to the quantity informed within the option below "Lines per page"
 - **Full** : This option will set the page to display all records in one page.
 - **Lines per page** : Here you can set the amount of records that are displayed in the detail form, you will need to set that just if paging is partial.
 - **When deleting** : This option allows you to choose how the form detail dependencies will be treated in a deletion. The options are:
 - **Warn when there are dependencies** : Displays an error message if there is dependency on the detail form record that is being deleted from the master form.
 - **Delete dependencies without warning** : Deletes all existing dependencies in the detail form without warning the user.
 - **Edit link** : This button will allow you to remake the link for application details.

Allows you to configure automatic update link tables in N:N relationships

Identification

This link will be handled as a field within the form application. We have to set the field name, data type, label and define if we will use a specific connection for this link. If decide to you use a specific connection there will be a combo box with for you to select according to the connections available in the project.

Field <input type="text"/>	Field Name.
Data Type <input type="text" value="Text"/>	Field data type.
Label <input type="text"/>	Application field title.
Connection <input type="checkbox"/> Choose connection	Check if you would like to use a different connection than the one of the application.

Creating field to update link table.

Grid Information

In this step you need to inform the data source, this connection will be available for the user to select (lookup). It allows you to choose if the select command is manually informed or if it will be based on a table.

Process that helps you to build your SQL Select Statement. You can fill the SQL Statement manually or choose the table.

Manually

Choose Table

Informing the select command.

If you check the option "Choose table" there will be 3 combo boxes for you to select according to the connections tables available in the project: table, key field and description.

Process that helps you to build your SQL Select Statement. You can fill the SQL Statement manually or choose the table.

Manually

Choose Table

Key

Description

Informing the select command based in a table..

- **ey** : Value that will be stored in the table.
- **escription** : Value that will be displayed in the update field.

SQL Select Statement

Select command informed by user or created based on a table, it is responsible for displaying the selection field contents.

SQL Select Statement from the foreign table that contains the data that will be used to update the relationship table. The SQL Statement must return 2 fields: the first is the key and the second is the description.

SQL Select Statement

```
select group_id, description from sec_groups order by description
```

Select command.

Lookup display

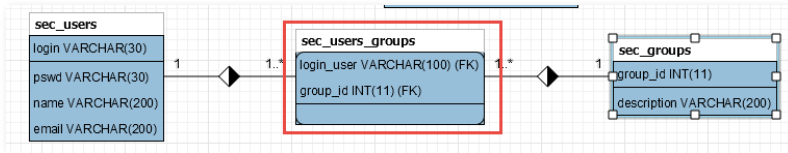
This option sets the display for the update field

Display key and description <input type="radio"/> Yes <input checked="" type="radio"/> No	For each record, display its key and description.
Display Title <input type="radio"/> Yes <input checked="" type="radio"/> No	Display attribute title.
Separator <input type="text"/>	Delimiter between key and description.
Object <input type="text" value="Double Select"/>	Lookup Object type.
Height <input type="text" value="7"/>	Object Height

Field object type.

- **isplay key and description** : If you want to display the code and the description informed in the select command.
- **isplay Title** : Displays attribute title (see attribute).
- **eparator** : When the option Display key and description is checked as yes you must define a separator between the code and the description.
- **bject** : Object field type that will be displayed in the update form. The types are: Select, Radio, Check box and Double Select.
- **heck and Uncheck All** : Option to Check and Uncheck all records (available only for when Check box object is selected).
- **olumns** : This option sets the number of columns (available only when the objects Check box or Radio is selected)
- **eight** : This option sets the object height (available only when the objects Select or Double Select is selected).

Link Table



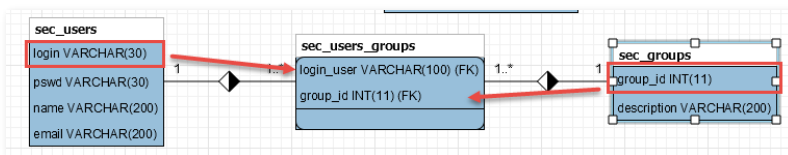
Link table.

This option sets the relationship table that will be updated.

Table Name
sec_users_groups

Update table selection.

Relationship Keys



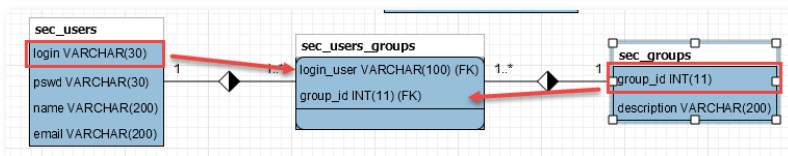
link key

This option sets the application fields that are related with the link table fields.

Field	Key
login_user	customerid
group_id	regionid

Setting values for the fields in the update table.

Lookup Field



Lookup field.

This option sets the field from the lookup table that is related to the link table.

Lookup field
group_id

Defining the foreign key relationship table..

Link Attributes

This option sets the values that are recorded in the fields of the link table that are not foreign key.

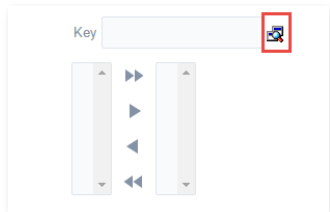
Field	Attributes
login_user	customerid

Setting the assignment field.

- **tributes** : It is written a value without the need for user data entry, it is an application field itself.
- **ata entry** : It is written a value reported by the user, it is necessary to create a field and relate it by the select.

Search

Available only for Double Select object



Search in the double select field.

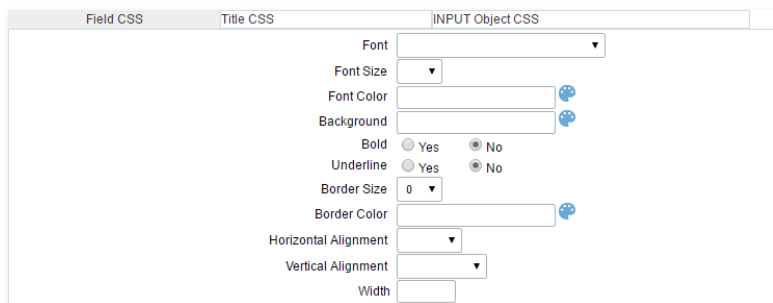
This option sets a filter for the link field.

![Search settings for the link field.][relacao_n_n_info_filtro_config] Search settings for the link field.

- **imit of Records** : Maximum number of records returned by the search.
- **nitial State** : this option sets whether the form field load come Filled or empty.
- **earch Fields** : This option sets whether the search will be displayed the code and/or Description.

General Display Settings

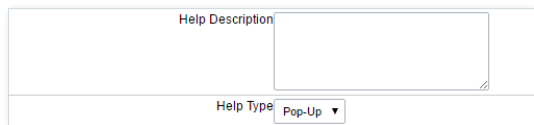
This option allows you to configure all general display settings from from field title and object.



Display settings.

Help Configuration

This feature allows it to be documented instructions for use of the application generated, that is, help end users to better understand the business rules of the system and the best way to operate it.



Help Configuration

- **tributes**
- **elp Description** : This option allows the inclusion of a text that will be displayed when the user to position the mouse over the field.
 - **elp Type**:
 - **op-up** : When choosing the type pop-up, help icon is displayed next to the field; that when clicked, will display what was reported in the attribute Describes Help.
 - **int** : When you pass the mouse cursor over the field, will help described in the field Describe Help.
 - **ext** : Next to the field will appear a text containing the help that was described in the field Describe Help.

Settings

With this interface, you can define general options of the Search Form.

SEARCH SETTINGS	
ATTRIBUTE	VALUE
Search Criteria	AND ▼
Display Condition	<input type="checkbox"/>
Use auto-complete in the fields.	Defined in the field ▼

Search configuration Interface.

Search Criteria

Allows to select the logical operator **AND** or **OR** to define the criteria of the search;

Display Condition

Gets the condition of the search available for the user to choose one. He can select "AND" or "OR" in a Combobox.

Use auto-complete in the fields

Automatically turns the field into an autocomplete according to the existing values in the database. If the user chooses **Yes**, the autocomplete will enable automatically in all inputs that contain a relationship. If the user decides **No**, so no autocompletes will be displayed. Otherwise, the option selected is **Defined in the field** it'll keep the settings for each field individually.

Dynamic Search

This feature allows the end-user to search for records without leaving the current screen. The developer must inform the available fields for search.

Settings

DYNAMIC SEARCH SETTINGS	
ATTRIBUTE	VALUE
Use the ENTER key to	▼

Dynamic Search Settings Interface

- **se the ENTER key to** : It allows you to define the action that the Enter Key has on the Search form. **Tabulate** enables you to navigate between fields, and **Submit** performs the search(activates the Search button).

Select Fields

This interface allows to select the available fields in the Dynamic Search.

SEARCH DYNAMIC FIELDS		
ATTRIBUTE	VALUE	DESCRIPTION
Select the fields that will be used for dynamic search.		
customerid companyname contactname contacttitle birthdate country regionid stateid city address postalcode phone fax cityid creditlimit cardtype cardnumber notes	⌘ ⤴ ⤵ ⌘	grid_customers

Dynamic Search Field Selection.

QuickSearch

Quicksearch is an option that allows to search data in various fields of the application by using the text box in the toolbar.

LIST - ORDERS 04/05/2017

All fields Quick search Columns Sorting Export Advanced Search Exit

Orderid	Customerid	Employeeid	Orderdate	Requireddate	Shippeddate
10,248	SANTG	4	07/04/2008	08/01/2008	08/10/2008
10,249	ANATR	7	07/05/2008	08/16/2008	07/10/2008
10,250	HANAR	4	07/08/2008	08/05/2008	07/12/2008
10,251	VICTE	3	07/08/2008	08/05/2008	07/15/2008
10,252	SUPRD	4	07/09/2008	08/06/2008	07/11/2008
10,253	HANAR	3	07/10/2008	07/24/2008	07/16/2008
10,254	CHOPS	5	07/11/2008	08/08/2008	07/23/2008
10,255	RICSU	9	07/12/2008	08/09/2008	07/15/2008
10,256	WELLI	3	07/15/2008	08/12/2008	07/17/2008
10,257	HILAA	4	07/16/2008	08/13/2008	07/22/2008

Go to 1 View 10 [1 to 10 of 831]

Application running QuickSearch.

Quick Search Settings

In the quick search settings you can find the following options:

ATTRIBUTE	VALUE
Maintain search condition	<input checked="" type="checkbox"/>
Button within the search	<input checked="" type="checkbox"/>
Quicksearch show combobox	<input type="checkbox"/>
Quicksearch Watermark	{lang_othr_qk_watermark}
Quicksearch width	10
Display the Quick search old format	<input checked="" type="checkbox"/>
Search anywhere	<input checked="" type="checkbox"/>

QuickSearch Configuration Interface.

Button within the search

It displays the search button inside the text area of the box.

Quick search show combo box

It displays a combo box if there is only one field option in the Quick search.

Quick search Watermark

It displays a hint text inside the Quick search.

Quick search width

It defines the width of the Quick search input.

Display the Quick search old format

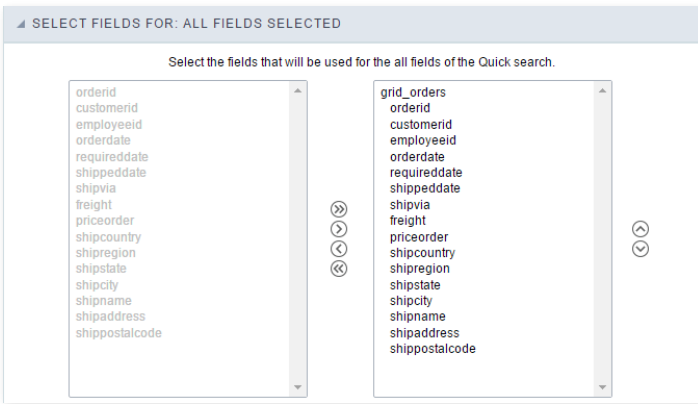
Displays in the old format with the field selection option in the Quick search.

Search anywhere

It allows searching each part of the String for the data-informed in the field.

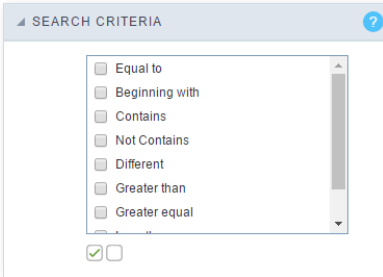
Select Fields

It allows selecting which fields the Quicksearch performs the searches.



QuickSearch selecting fields Interface.

And you can select various criteria of the search.



QuickSearch search criteria configuration Interface.

You need to add the QuickSearch button in the toolbar to use it.

Fields

This is a list of fields available for use in the filter. Click on the desired field to access the configuration instructions.

Text Field	Field Select
Integer Field	Field Double Select
Decimal Field	Field Checkbox
Currency Field	Field Radio
Date Field	Field Text Auto-Complete
Field time	Field Number Auto-Complete
Field Datetime	

Settings

With this interface, you can set the common attributes of the app.

ATTRIBUTE	VALUE
Application Code	grid_customers (9.00.0000)
Description	<input type="text"/>
Documents Path	C:/Program Files/NetMake/v9/www/
Image Directory	/scriptcase/file/img
Application images	<input type="button" value="Add"/> <input type="button" value="Remove"/>
Language	English (United States) ▼
Share Location Variable	<input checked="" type="checkbox"/>
Charset	<input type="text"/>
Share Theme Variable	<input checked="" type="checkbox"/>
Folder	root ▼
Edit by Project	<input checked="" type="checkbox"/>
Timeout	<input type="text" value="0"/>
HelpCase Link	Application ▼ Search ▼ Summary ▼

Application Settings Interface

- Attributes**
 - Application Code** : It is the name that defines an application. An app can be renamed at the [List of Application](#).
 - Description** : This field contains a brief description of the application objectives.
 - Documents Path** : The absolute path to store uploaded documents in the application.
 - Image Directory** : The filesystem directory to store the application images.
 - Application images** : Import images into the application to allow using them in the application.
 - Language** : Set the default language of the application. Display all the application hints and messages in the selected language.
 - Share Location Variable** : Define if the app shares the regional settings with other applications through a session variable.
 - Charset** : Define a specific charset to use in the application.
 - Share Theme Variable** : Define if the app shares the Theme settings with other applications through a session variable.
 - Folder** : Define the project folder that contains the app.
 - Edit by Project** : Define if other project developers can edit the application.
 - Timeout** : Set the session runtime timeout in seconds. If the value is Zero, it assumes the default timeout of the PHP.
 - HelpCase Link** : It allows to associate a [HelpCase](#) file with the application.

Notification Settings

ATTRIBUTE	VALUE
Use SweetAlert	<input type="checkbox"/>
Error Position on the field	Down ▼
Show the Error Title in the Application	<input checked="" type="checkbox"/>
Show the Error Title in the Field	<input type="checkbox"/>
Error Title	{lang_errm_errt}
Script Error	<input type="checkbox"/>
SQL Error	<input checked="" type="checkbox"/>
Debug Mode	<input type="checkbox"/>
Ajax Error Output	<input checked="" type="checkbox"/>

- Use SweetAlert**: Use the SweetAlert to display messages from the application. When this option is active, it will replace the browser's "confirm" and "alert".
- Use SweetAlert position using Toast** : The position to display error messages on the application.

- **error Position on the field** : The position to display error messages when criticizing the field.
- **how the Error Title in the Application** : Define to display the title line of the error message or not.
- **how the Error Title in the Field** : Define to display the title line of the error message in the field or not.
- **cript Error** : Allows displaying the line code where there is an error..
- **QL Error** : Allows displaying the SQL statement if it got an error.
- **ebug Mode** : Runs the application in Debug mode, showing all SQL statements the application is executing.
- **jax Error Output** : Enables the Ajax alert for debugging errors.

Navigation

This interface allows defining the navigating behavior of the application

NAVIGATION	
ATTRIBUTE	VALUE
Exit URL	<input type="text"/> 🔗
Close on Exit	<input type="checkbox"/>
Redirect URL	<input type="text"/> 🔗
Redirect Variable	<input type="text"/>

Navigation Interface.

Exit URL

URL to where the user goes when he clicks on the “exit” button.

Close on Exit

Close the browser window when the user clicks on the “exit” button.

Redirect URL

Redirect to another URL in case there aren't any global variables available.

Redirect Variable

Creates a variable with the application name and sends it to the redirected application.

Messages

On this screen, you can redefine the default application messages to the end-user by customizing the validation messages for Insert, Update, and Delete. You can also define messages for SQL errors and the confirmation outputs.

Messages

Messages		
ATTRIBUTE	VALUE	DESCRIPTION
No Records Message	<input type="text"/>	When the application has no records.it will display this customized text.
Primary key violation	<input type="text"/>	Message to display when the primary key constraint is violated
Unique key violation	<input type="text"/>	Message to display when the unique constraint is violated

No Records Message

The message displayed when the application has no records.

Primary key violation

The message displayed when there is a violation in the database's primary key constraint.

Unique key violation

The message displayed when there is a violation in the database's unique constraint.

Insertion Messages

Insertion messages		
ATTRIBUTE	VALUE	DESCRIPTION
Message After Insert	<input type="text"/>	Message to display after inserting a record.
Message to Confirm Insert	<input type="text"/>	Message to display to confirm inserting of a record.

Message After Insert

The message displayed when inserting a new record.

Message to Confirm Insert

The message displayed if the end-user wants to confirm the inserting of a new record.

Update Messages

Update messages		
ATTRIBUTE	VALUE	DESCRIPTION
Message After Update	<input type="text"/>	Message to display after updating a record.
Message to Confirm Update	<input type="text"/>	Message to display on the update confirmation

Message After Update

The message displayed when updating a record.

Message to Confirm Update

The message displayed if the end-user wants to confirm the changes of a record.

Messages Of Delete

Messages of Delete.		
ATTRIBUTE	VALUE	DESCRIPTION
Message After Delete	<input type="text"/>	Message to display after delete a record.
Message to confirm Delete	<input type="text"/>	Message to display on the delete confirmation

Message After Delete

The message displayed when deleting a record.

Message to confirm Delete

Displays a customized message asking to confirm the record deletion.

If you make no changes, the application will use the default values for the messages. Those values can be configured in [Locales -> Application Language](#).

Global Variables

This screen shows all global variables used in the application

The global variable is an external parameter required for running the application. The application can include global variables in the WHERE clause, field definitions and names, event programming, and so on.

You can define global variables by using square brackets ([variable]). You can pass the parameters to the application through one of the methods: Session, Post, and Get

if you need to use [Database Identifiers](#) in or tables, we recommend using the double quotation marks(" ") instead of the brackets([]) because of the conflict with the [Global Variables syntax](#) of Scriptcase. Using the brackets as database identifiers may lead to problems in the Scriptcase applications.



Global variables Interface.

Example:

```
Select CustomerID, CustomerName from Customers WHERE CustomerID = '[v_customerid]'
```

We do not recommend using session variables (global variables) with the same name as the table fields.

In this case, the variable **v_customerid** is displaying in the global variables configuration.

The screenshot shows a configuration window titled 'VARIABLE SETTINGS'. It has two main columns: 'ATTRIBUTE' and 'VALUE'. Under 'ATTRIBUTE', the value 'global' is entered. Under 'VALUE', there are three sections: 'Scope' with checkboxes for 'SESSION' (unchecked), 'POST' (checked), and 'GET' (checked); 'Settings' with a checkbox for 'Optional' (unchecked); and 'Type' with radio buttons for 'Out' (unchecked) and 'In' (checked).

Global variables configuration Interface.

Attribute

It shows the variable's name in the application.

Value

Defines the behavior of the variables, divided into three blocks, they are:

Scope

Defines how the application receives the variable. A variable defined as the POST method in the app does not receive value if it comes with the GET method. Those are the methods available:

- **ession**: Defines that the variable must be created as a PHP session variable by another application.
- **ET**: Defines that the variable must be passed by the URL, that is, visible by the browser.
- **OST**: Defines that the variable must be passed through the POST method of an HTML form.

Settings

- **ptional**: Check it to don't validate the variable at runtime.

Type (In / Out)

Defines whether the variable comes from another application (In), or it is starting in the current app (Out).

Description

Indicates where the application is using the variable.

Synchronize table

This process performs a comparison between the definitions of application fields and fields of the connected database table. If there is any change within the connected table, such as adding a new field, exclusion, or a change in the data type, the changes will appear visible as in the image below:

Application Fields	Table Fields
customerid	customerid
companyname	companyname
contactname	contactname
contacttitle	contacttitle
birthdate	birthdate
country	country
regionid	regionid
stateid	stateid
city	city
address	address
postalcode	postalcode
phone	phone
fax	fax
cityid	cityid
creditlimit	creditlimit
cardtype	cardtype
cardnumber	cardnumber
notes	Files

■ Fields that will be created
■ Fields that will be updated
■ Fields that will be removed

Table synchronization interface.

After accessing the function from the link "Synchronize table" you will see a comparison table between the application fields (left table) and the database table fields (right table), above image, the fields highlighted in "red" will be deleted of the application, the fields highlighted in "Green" will be inserted in the form, and the fields highlighted in "orange" will be updated, so data type will be updated.

To rename a field in the database table, the table synchronization effect, by comparison, will be the same as deleting a field that existed in the application and the inclusion of a new field.

Below you can see a video showing the process:



1. [Home](#)
2. Applications

General Overview

Allows to create link between applications that are in the same project, the link option incorporates various functionalities like above.

New Link

To create a new link you just need to access the option **Link between Applications** located in the menu at the left side.

When clicking in **New Link** the screen below will be display:

When clicking in **New Link** the screen to choose the type of link that will be created will be displayed.

[Edit Link:](#)

Allows to edit registers of the grid using a form.

[Capture Link:](#)

Allows to create a link specifically to the filter's fields, capturing data from other grid.

[Field Link:](#)

Allows to create a link represented through a link, from a grid's column to any application from the project.

[Button Link:](#)

Allows to create a link through grids buttons to any application from the project.

1. [Home](#)
2. [Applications](#)
3. Form
4. Application link

Application link

Creating an Application Link

This type of link allows the developer to create a link from a grid to a form with the objective of editing the register of a grid's row.

In the link options, we will choose the **Application Link**. When we choose this option, edit a register from a Grid will be possible.

Applications list

After we select this option, will be displayed the applications list to what you want to create the link.

This screen can be viewed from the following ways:

All:

In which is possible to see all the project's applications. **Example:**

By folder:

In which is possible to see the applications according to the folders in which each of them are.

By type:

In which is possible to see the applications grouped by its respective types.

Definition of Parameters

In this setting we are going to define the type of parameter that will be passed to the next application.

We have two options at **Type of Parameters**, they are:

Fixed value:

This option allows the user to define a fixed value that will be used to the call of the next application.

No value:

This option allows to create the link without the need to send any parameter to the next application.

It is also possible to see a refresh button at its right side.

This button should be used when a new parameter is added to the target application, so the new parameter can be loaded in the current application to make the link.

Form Properties

In those settings, we can define which buttons will be available in the target form application. Initially we have five options, that are:

Redirect after inclusion:

In this option we can define what will happen with the application after insert the record.

Redirect after update:

In this option we can define what will happen to the application after update the record.

1. [Home](#)
2. [Applications](#)
3. Form
4. Capture link

Capture link

Applications list

After we select this option, will be displayed the applications list to what you want to create the link.

This screen can be viewed from the following ways:

All:

In which is possible to see all the project's applications. **Example:**

By folder:

In which is possible to see the applications according to the folders in which each of them are.

By type:

In which is possible to see the applications grouped by its respective types.

Definition of Parameters

In this setting we are going to define the type of parameter that will be passed to the next application.

We have two options at **Type of Parameters**, they are:

Fixed value:

This option allows the user to define a fixed value that will be used to the call of the next application.

No value:

This option allows to create the link without the need to send any parameter to the next application.

It is also possible to see a refresh button at its right side.

This button should be used when a new parameter is added to the target application, so the new parameter can be loaded in the current application to make the link.

Application with no parameters

However, when the target application does not have any defined parameters, the following screen is displayed:

![Without parameters][semparametro]

Clicking in the button, you will be taken to the target application to create a parameter, so you can use the update button in the **parameters definitions** to refresh them.

Link properties

In this screen we will set the application display mode that will be called in the link.

In this type of link there is only one display option:

Modal:

When we use this option the target application will be displayed in the same window of our application, however it will be displayed in a small window above the current application.

Modal

When we use this option the target application will be displayed in the same window of the application, however she will be displayed in a small window above the current application.

When we select this option, we can set the following options:

Modal:

In this option we are going to define if the target application will be opened in a Modal.

Yes: This option will make the target application be opened in a modal. **No:** This option will make the target application be opened in a new window.

If **Yes** is selected in the previous option, the Modal **Height** and **Width** will be available.

Height:

Allows to set the Modal height.

Width:

Allows to set the Modal width.

If **No** is selected in the previous option, only those options will be available.

Allows to modify manually in the update:

When we select the option **Yes**, the button **New** is created in the grid to insert new registers.

Allows to modify manually in the insert:

Indicates which shortcut key to the button add new register.

Allows to modify automatically in the update:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Allows to modify automatically in the insert:

Choosing the option **Yes**, the window will be closed after doing the insert of a register.



1. [Home](#)
2. [Applications](#)
3. Form
4. Link edit

Link edit

In the Link Folder of the Application Menu (Image Below) are displayed the links existing in the application and also the item New Link. When clicking on the existing link it is displayed the screen below that allows to manage the links.

 *Editing Links.*

Actions

Properties Change the link's behavior, position, and how the link opens.

Link Change the application that's being called in the link their parameters.

Delete Remove the existing link.

1. [Home](#)
2. [Applications](#)
3. Form
4. Button link

Button link

Creating a Button Link

Allows the developer to create a link where the call to the other application will be done through a button.

In the type of links options, we will choose the **Button Link**. Choosing this option it will be possible to create a link to any other application.

List of applications

After selecting this option, The list of applications to what you want to create a link will be displayed.

This screen can be viewed in the following ways:

All:

In which is possible to see all the project's applications. **Example:**

By folder:

In which is possible to see the applications according to the folders in which each of them are.

By type:

In which is possible to see the applications grouped by its respective types.

Definition of Parameters

In this setting we are going to define the type of parameter that will be passed to the next application.

We have two options at **Type of Parameters**, they are:

Fixed value:

This option allows the user to define a fixed value that will be used to the call of the next application.

No value:

This option allows to create the link without the need to send any parameter to the next application.

It is also possible to see a refresh button at its right side.

This button should be used when a new parameter is added to the target application, so the new parameter can be loaded in the current application to make the link.

Link properties

In this screen we will set the application display mode that will be called in the link.

There are five display options, they are:

Open in the same window:

When we use this option the target application will be displayed in the same window of the application, and the target application will have a back button so we can return to the previous application.

Open in another window:

When we use this option the target application will be displayed in other browser window, and the target application will have a exit button so we can close this window.

Modal:

When we use this option the target application will be displayed in the same window of our application, however it will be displayed in a small window above the current application.

Open in the same window

When we use this option the target application will be displayed in the same window of the application, and the target application will have a back button so we can return to the previous application.

When we select this option, we can set the following options:

Exit URL for the target application:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Hint for the link:

In this option we will inform a message that will be displayed when the mouse cursor is over the button with the link.

Open in another window

When we use this option the target application will be displayed in another browser window.

When we select this option, we can set the following options:

Exit URL for the target application:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Close the form after an update:

Choosing the option **Yes**, the window will be closed after doing the update of a register.

Close the form after an insert:

Choosing the option **Yes**, the window will be closed after doing the insert of a register.

Hint for the link:

In this option we will inform a message that will be displayed when the mouse cursor is over the button with the link.

Open the application in a tab inside the menu:

Choosing the option **Yes**, the application will be open in **amenu tab**, when executed from a menu application, instead of opened in another browser window.

Choosing the option **Yes**, there are also some other settings to be displayed:

Title to the tab:

This option allows the developer to set a title to the tab that will be opened when used in a Menu application.

Hint to the tab:

This option allows the developer to set a message to be displayed when the mouse cursor is over the tab Menu.

Active tab icon:

This option allows the developer to set an icon to be displayed in the tab when used in a menu application.

Modal

When we use this option the target application will be displayed in the same window of the application, however she will be displayed in a small window above the current application.

When we select this option, we can set the following options:

Height:

Allows to set the Modal's height.

Width:

Allows to set the Modal's width.

Exit URL for the target application:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Close the form after an update:

Choosing the option **Yes**, the window will be closed after doing the update of a register.

Close the form after an insert:

Choosing the option **Yes**, the window will be closed after doing the insert of a register.

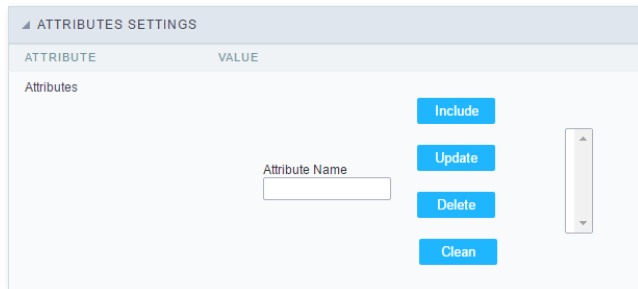
Hint for the link:

In this option we will inform a message that will be displayed when the mouse cursor is over the button with the link.

ScriptCase has incorporated the concept of Object Oriented programming, using attributes, resources, methods and libraries. It is possible to create your own business rules in applications, and by using these concepts you can reap huge rewards in terms of better organization and improved development.

Attributes

The attributes are variables that has the global scope declared in the application. An attribute that can be referenced in all the methods and the application's events.

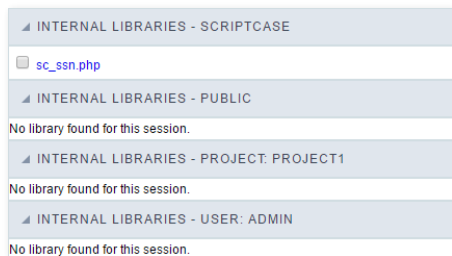


Attributes configuration Interface.

- **Attributes**
 - **Attribute Name** : This field allows to inform the attributes name, after typing the name just click on **include**. In case you want to **update** click on the attributes name (located on the buttons' right side), do the necessary modifications and click on update. In case it is necessary to **delete** an attribute, it is necessary to do same process as the update, but when selecting the attribute, it is necessary to click on the delete button. The **clear** button, as the name suggests, it clears the list of all the attributes listed on the right.

Internal libraries

To use the internal libraries just select the correspondent (Project Libraries for example). This process allows the methods existing in the library, to be visible in all the application's Events and Methods. See how to manage the libraries by [clicking here](#).

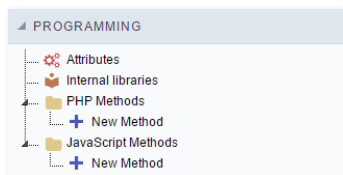


Internal Libraries management Interface

PHP Methods

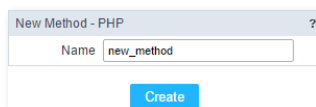
Methods are function or procedures declared by the developer, that helps when applying the routines. Using methods in the applications allows to reuse your code throughout the application, optimizing the development experience.

Creating A New Method



- Give the method a name and click Create. As in the image below.

PHP methods are incorporated into the same class, so it is not possible to use two functions with the same name even when used in different methods.



- Methods can receive parameters.

```
function new_method
1 echo "Hello World!!!";
```

- dd the amount of variables:

Definition of the parameters of the method:

new_method

- No defined parameter.

Add 1 Parameter(s) Cancel

Save

- refining the variables:

Insertion of Parameters

Name	Type	Value Standard
	For Value	
	For Value	
	For References	

Save Back Cancel

- **ame** : Type in the variable's name.
- **ype** : Selecting the type of variables: For Value or For Reference.
- **alue Standard** : The parameter's value used to initialize when calling the method.

References allows to create a second name for a variable that you can use to read and modify the variable's original information.

- diting a parameter:

Definition of the parameters of the method:

new_method

Parameters
Stest = test

☑ ☐ ✎ ✕

Add 1 Parameter(s) Cancel

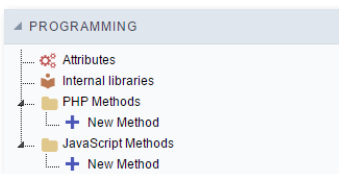
Save

- ☑ : Checking all the parameters in the parameter list.
- ☐ : Uncheck all the parameters in the parameter list.
- ✎ : Edit the selected parameter of the list.
- ✕ : Deletes the selected variable of the list.

JavaScript Methods

Methods are function or procedures declared by the developer, that helps when applying the routines. Using methods in the applications allows to reuse of the your code throughout the application, optimizing the development experience.

Creating A New Method



JavaScript method creation Interface

- refine a name for the method and click on Create. Like the image below.

New Method - JAVASCRIPT ?

Name

Include Method.

- methods can receive parameters.

function new_method

```
1 echo "Hello World!!!";
```

Definition of the parameters of the method:

new_method

No defined parameter.

Parameter(s)

- add the amount of variables:

JavaScript method parameter definition

- defining the variables:

Insertion of Parameters

Name	Type	Value Standard
<input type="text"/>	For Value	<input type="text"/>

Defining the variables

- **name** : Type in the variable's name.
- **type** : Selecting the type of variables: For Value or For Reference.
- **value Standard** : The parameter's value used to initialize when calling the method.

References allows to create a second name for a variable that you can use to read and modify the original variable's information.

- editing a parameter:

Definition of the parameters of the method:

new_method

Parameters

Stest = test

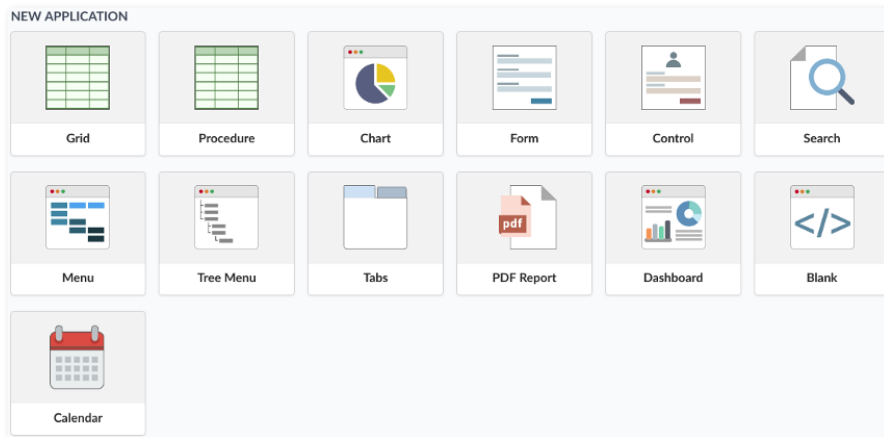
Parameter(s)

- : Checking all the parameters in the parameter list.
- : Uncheck all the parameters in the parameter list.
- : Edit the selected parameter of the list.
- : Deletes the selected variable of the list.

New Application

The creation process of one application is simple enough and intuitive.

You just need to select which application you want to generate in the applications available in Scriptcase, That it will be generated automatically.



Application Data

When the application is selected, the following screen will be displayed below.

In this table you need to select the connection that will be used in the application. The tables will be loaded and displayed in the field **Table** when the connection is selected.

It is possible to select more than one table in the application creation.

- **connection** - It defines the used connection to create the application. All the project's application will be displayed.
- **ables** - It defines the used tables in the application. (Form and Calendar can only use one table).
- **ields** - It defines the fields that will be part of the applications.
- **QL Select Statement** - Display the select statement created after selecting the tables and fields. This field allows also to insert a previously created SQL, if the used tables are available in the database selected in the connection.

Edit Fields

In this tab will be displayed the fields that are going to be created in the application, also its labels and datatypes. We can make define some settings here like changing its labels and datatypes.

Theme

In this screen you select the theme that the application will use. The default project's theme, that can be checked and changed in **Project > Properties**, will be selected automatically.

APPLICATION DATA RELATIONSHIP EDIT FIELDS **THEME**

Sc9_Rhino ▾

Header

◀ ◀ ▶ ▶ Add Save

Block 1.1

Title 1 Object text


Block 2.1

Title 1	Title 2	Title 3
11111111	22222222	33333333
11111111	22222222	33333333

Footer

- **heme** - List of available themes in the project.

*After all the settings are done, click in the button **Create**. In the next screen the settings of the created application will be displayed.*

TAB SETTINGS	
ATTRIBUTE	VALUE
Title	<input type="text" value="tabs"/>
Friendly URL	<input type="text"/>
Organization	<input type="text" value="Iframe"/>
Type	<input type="text" value="Folder on the Top"/>
Table Width	<input type="text" value="80"/>
Measure Unit	<input type="text" value="Percent"/>
Tab Alignment	<input type="text" value="Left"/>
Application alignment	<input type="text" value="Center"/>
Margins	<input type="text"/> Up <input type="text"/> Down <input type="text"/> Right <input type="text"/> Left
Exit	<input type="text" value="Button on the bottom of the page"/>
Exit Icon	<input type="text"/> 

Tab settings interface.

Within the Tab settings you can configure the following attributes:

- **itle** : Application title that will be displayed in the toolbar.
- **riendly URL** : This field allows you to customize the URL that will be called by the application, the allowed characters are the same available on regular URLs: a-z, A-Z, 0-9, -_. This option can also be changed within Scriptcase Home interface, on the "Friendly URL" column from the applications grid.
- **rganization** : Allows the applications to run inside folders via iframe or embedded on the tabs application.
- **ype** : This option sets how the folders will be organized, it can be:
 - **older on the Top** : Displays the folders above the application.
 - **older on the Bottom** : Displays the folders below the application, next to the footer.
 - **older on the Left** : This option allows the folders to be aligned in the left, like a vertical menu.
 - **older on the Right** : This option allows the folders to be aligned in the right, like a vertical menu.
 - **enu on the Top** : This option is similar to "Folder on the Top", but its visual schema has the menu style.
 - **enu on the Bottom** : This option is similar to "Folder on the Bottom", but its visual schema has the menu style.
 - **enu on the Left** : This option is similar to "Folder on the Left", but its visual schema has the menu style.
 - **enu on the Right** : This option is similar to "Folder on the Right", but its visual schema has the menu style.
 - **ide by Side** : Display all applications in folders, one next to the other.
- **able Width** : Main table size.
- **easure Unit** : The unit used by table size. It can be: Pixel or percentage..
- **ab Alignment** : This option sets the tab alignment position. It can be: center, left or right.
- **pplication alignment** : This option sets the application alignment position, It can be: center, left, or right.
- **xit** : This option allows you to define how the exit will be presented in the application, the options:
 - **utton on the bottom of the page;**
 - **older;**
 - **utton in the toolbar**
 - **n view.**
- **xit Icon** : This option allow you to select an icon for the "Exit" option in the tab. Use the image manage to select or import a customized image.

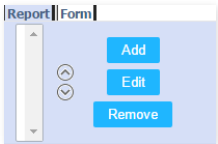
You will use the folder settings options to manage each folder link to an existing application.

The screenshot shows the 'FOLDER SETTINGS' interface. On the left, there is a list of folders with 'Report Form' at the top. To the right of the list are navigation arrows (up, down, refresh) and a 'Clear' button. The main form area has a 'Label' input field, an 'Image' input field with a small image icon, and buttons for 'Add', 'Update', and 'Remove'. At the bottom, there are two input fields: 'Minimum tab width' and 'Maximum tab width', with explanatory text for each.

Folder Settings Interface.

- **add a folder** : To create a tab, click in 'Add' and fill these fields: * **Label** : Application name that will be displayed in the tab or menu (examples: Personal Data, Finance Data, etc.); * **Image** : Choose the image that will be used for this tab;
- **update** : Double click on the folder name (left column / list of folders) and then update the data. Finally, click on the 'update' button.
- **delete** : Double click on the folder name (left column / list of folders) and then delete the item.
- **remove** : This button when clicked, clears all existing folders in the box (the box located in the left).

In this folder you can set which applications will be part of the tabs implemented, as follows:



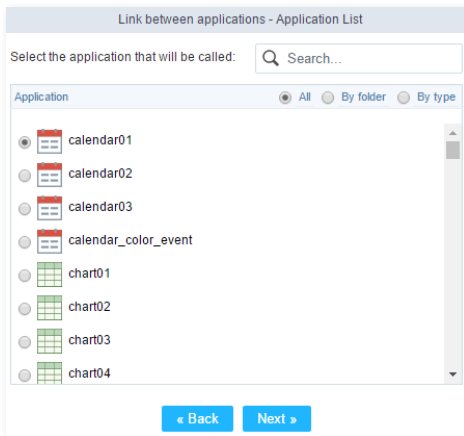
Application Settings Interface

Initially you must click on the tab that you need to update. You have the options:

- **add** : Click "Add" button, and then the following screen will be displayed for selection of applications.
- **edit** : Select the application for the link and then click the "Edit" button. The following steps are the application selection and setting the size of the iframe, as requested in the process to add applications.
- **remove** : To remove just select the application you want to delete in the folder, and then click "Remove".

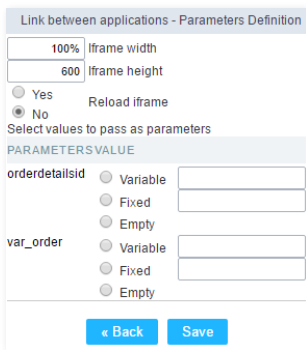
Adding And Editing Applications

Select the application to be placed in the folder, and click the continue button, you must see the screen below:



Application Link Interface.

Then the following screen will be displayed to complement with the link parameters to the selected application:



Parameters set Interface..

- **frame width** : Width set to be used inside the folder to display the application window, leaving 100% the application will be adjusted more easily.
- **frame height** : Height set to be used in the iframe to display the application window, if the it is too small, the application will create a scroll bar to the window.
- **eload iframe** : When a folder is selected the applications contained in the same run and the result is displayed, with this option turned on "Yes" when you click on the folder that contains the application cache is created with the page loaded, this cache will be used if the user browse for other folders and return to the folder that has the application with this option enabled.
- **arameters / Value** : For each parameter of the application we are putting in the folder, there are three options to choose:
 - **ariable** : Global variable name where the content will be retrieved.
 - **alue** : Content to be loaded into the parameter at run time.
 - **mpty** : For this parameter the content will not be the responsibility of the Tab application

ATTRIBUTE	VALUE
Create PDF	<input checked="" type="checkbox"/>
PDF Orientation	Portrait
PDF Format	Letter (216 x 279 mm)
Open PDF Directly	<input type="checkbox"/>

PDF Settings Interface.

- **reate PDF** : This option allows to define if the application will have output in PDF format.
- **DF Orientation** : This option sets the direction of the reports generated in PDF, it can be: portrait or landscape.
- **DF Format** : Pages format of the reports generated in PDF, such as Letter, A4, A5, etc.
- **pen PDF Directly** : This option indicates whether the application will allow us to generate the PDF directly via button or through a link to the PDF.

Security ?		
ATTRIBUTE	VALUE	DESCRIPTION
Use Security	<input type="checkbox"/>	Application needs validation to run. Read about the sc_apl_status macro.
Url output of the security	<input type="text"/>	Uri to be redirected if the user does not have access to this application.
Use Password	<input type="checkbox"/>	Enter password required to run the application.
Request password just once	<input type="checkbox"/>	Request password just once per session without asking for password every time you open the application in the same session.
Enable direct call by URL	<input checked="" type="checkbox"/>	Allow an application to be called by typing the URL directly in the browsers.

Application Security Configuration

Use Security

Turns the application accessible just when using the macro sc_apl_status or using the security module. When it is enabled, the application displays a message "User not Authorized" if the user tries to access it directly.

Url output of the security

It redirects the application to another URL after displays the warning "User not Authorized".

Use Password

Turns the application accessible just by using a password.

Request password just once

Request the password once per session, not needing to inform the password every time you access the application.

Enable direct call by URL

Allows to call the application directly from the URL in the browsers.

Allows calling the application directly from the URL in the browsers. In the development environment, you can disable those options for testing, it avoids requiring the login every time you run the application. To disable the security and password in development, you must access Options > My ScriptCase and uncheck the options: Enable Use of Security, Enable Use of Password.

This interface allows you to define a Log schema to the app. The Log schema tracks what the end-users are doing in the app. Those are the events it can tracks: Access, Insert, Update, and delete.

To learn how to create a Log schema [see here](#).

The screenshot shows a configuration window titled "LOG". It features a table with two columns: "ATTRIBUTE" and "VALUE". The "Schema" attribute has a dropdown menu currently set to "log". Below the table, there are two empty list boxes: "Events" on the left and "Access" on the right. Between these two boxes are two blue buttons: "Add >>" and "Remove <<".

Application Log configuration

Schema Select an existing log schema for the application.

Events You can select just specific events to save in the log.

This module lets you editing display options for the application. You can apply visualization themes, organize blocks, define values, and the display format of the Header and Footer.

Layout Settings

ScriptCase creates an app by using default values defined per project, but you can change those values for a specific application on this interface.

LAYOUT SETTINGS

Header Template	<input type="text" value="Flat"/>	Template name used for the application header.
Footer Template	<input type="text" value="Default"/>	Template name used for the application footer.
Button	<input type="text"/>	Use different buttons than what was defined in the color scheme.
Themes	<input type="text" value="Sc9_Rhino"/>	Use different themes from the one defined for the color scheme

Header

|< < > >| xxyyzz xxxxx yyyyy ▾

Block 1

Name

Type Male Female

Address*

Groups* Male Female

Countries

Address

Photos

Drag & Drop files here

Image1.png ✓
Image2.png ✗

Captcha

Application Layout Settings

Header Template

Allows choosing the look of the application Header.

Footer Template

Allows choosing the look of the application Footer.

Button

Allows choosing the button theme for the application.

Themes

Choose one of the selected themes in the project properties. It defines the look of the application, like colors, fonts, and others.

Header & Footer

Header

Here we can define if we want to display the header or not, the application title, and fill the header variables. The header looks according to the selected template in the Layout settings interface.

HEADER	
ATTRIBUTE	VALUE
Display Header	<input checked="" type="checkbox"/>
Insertion Title	{lang_othr_frmj_tit} - orders
Update Title	{lang_othr_frmu_tit} - orders
Header Variables	
SC_HEADER	Title
SC_VALUE	Date

Application Header Configuration

Display Header


This option determines if we want to display the header or not.

Title

Allows informing the title of the application.

Header Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **itle**: It displays the value of "**Application Title**" in the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Depending on the Application, you may have more than one title option.

Footer

FOOTER	
ATTRIBUTE	VALUE
Display Footer	<input type="checkbox"/>
Footer Variables	
NM_FOOTER_DATA	
NM_FOOTER_LOGO	
NM_FOOTER_TITLE	

Application Footer Settings


This page may change depending on the footer template chosen in the Layout Settings.

Display Footer

This option determines if we want to display the footer or not.

Footer Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

General data

SETTINGS	
ATTRIBUTE	VALUE
Application Code	tabs (9.00.0000)
Connection	Not defined ()
Default Profile	
Language	English (United States)
Edit by Project	<input checked="" type="checkbox"/>
Share Location Variable	<input checked="" type="checkbox"/>
Charset	
Share Theme Variable	<input checked="" type="checkbox"/>
Exit URL	
Folder	root
Description	
HelpCase Link	

Settings interface.

In this folder you can set the application configuration attributes, such as:

- **pplication Code** : Code for internal use of ScriptCase. Should start by a letter.
- **onnection** : Connection name of the used by ScriptCase to access the tables. The user can select it.
- **efault Profile** : Default profile to be loaded into the production environment.
- **anguage** : Data formatting based on the regional settings.
- **harsset** : Share location with other applications based on the values in the session.
- **hare Theme Variable** : Shares themes with other applications based on the values in the session.
- **xit URL** :URL to which the user will be redirected to exit the application.
- **older** : Folder where the application will be stored in the work group. The user can select it.
- **escription** : Application description.
- **elpCase Link** : Using this option you can associate HelpCase files to your application.

Global variables

This screen shows all global variables used in the application

The global variable is an external parameter required for running the application. The application can include global variables in the WHERE clause, field definitions and names, event programming, and so on.

You can define global variables by using square brackets ([variable]). You can pass the parameters to the application through one of the methods: Session, Post, and Get

if you need to use [Database Identifiers](#) in or tables, we recommend using the double quotation marks(" ") instead of the brackets([]) because of the conflict with the [Global Variables syntax](#) of Scriptcase. Using the brackets as database identifiers may lead to problems in the Scriptcase applications.

APPLICATION	
Settings	
Navigation	
Messages	
Global Variable	

Global variables Interface.

Example:

```
Select CustomerID, CustomerName from Customers WHERE CustomerID = '[v_customerid]'
```

We do not recommend using session variables (global variables) with the same name as the table fields.

In this case, the variable **v_customerid** is displaying in the global variables configuration.

Global variables configuration Interface.

Attribute

It shows the variable's name in the application.

Value

Defines the behavior of the variables, divided into three blocks, they are:

Scope

Defines how the application receives the variable. A variable defined as the POST method in the app does not receive value if it comes with the GET method. Those are the methods available:

- **ession**: Defines that the variable must be created as a PHP session variable by another application.
- **ET**: Defines that the variable must be passed by the URL, that is, visible by the browser.
- **OST**: Defines that the variable must be passed through the POST method of an HTML form.

Settings

- **ptional**: Check it to don't validate the variable at runtime.

Type (In / Out)

Defines whether the variable comes from another application (In), or it is starting in the current app (Out).

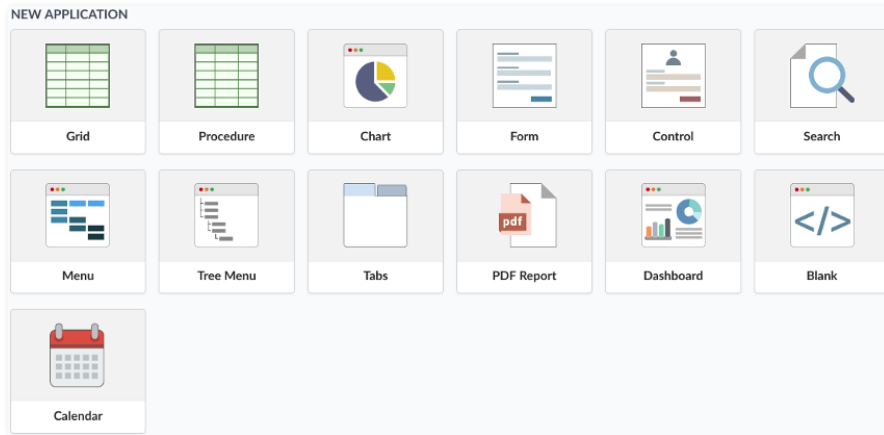
Description

Indicates where the application is using the variable.

New Application

The creation process of one application is simple enough and intuitive.

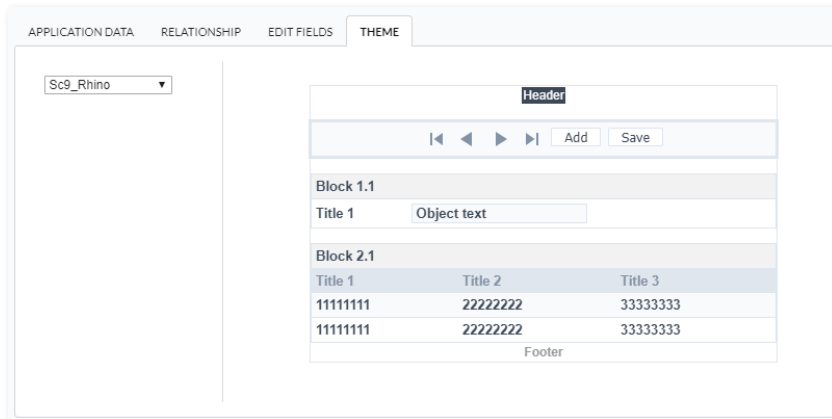
You just need to select which application you want to generate in the applications available in Scriptcase, That it will be generated automatically.



- **ame** - Application's name that is being created, It can not contain special characters.
- **ocalization** - Application's Language. The project's default language will be automatically selected.

Theme

In this screen you select the theme that the application will use. The default project's theme, that can be checked and changed in **Project > Properties**, will be selected automatically.



- **heme** - List of available themes in the project.

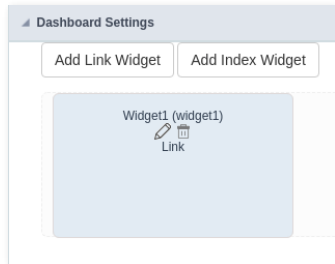
*After all the settings are done, click in the button **Create**. In the next screen the settings of the created application will be displayed.*

General Overview

The Dashboard application is focused on creating components where you can visualize several applications at the same time.

With the dashboard application, it is possible to create, for example, a screen to visualize several graphs, and also, with the addition of the Index Widget create comparisons, show sum of values, among other functionalities.

On the home screen, you can also configure where the Widgets will be displayed, they can be easily dragged and mounted according to the user's need.



At the homepage you can see the Widgets and some options.

Add Link Widget

This is the option to create a [link widget](#)

Add Index Widget

This is the option for creating an [index widget](#)


And within these, there are the options for:

Pencil

Used to edit Widget settings.

Trash

Used to Delete a Widget.

Besides that when the mouse is over the Widget, we can see a button  to change the size of it.

Add Link Widget

This option allows any application that exists in the project to be added to a Widget.

Link
Toolbar
Properties

Enter here the link which will open within the widget. It can be a Scriptcase application's name or an external URL. When you select an application, if it has links to other applications, you will be able also to set where these application links will opened.

Link

Link

To create the link enter the Widget and the application, it is necessary only to click on the field "Connection", then it will be shown a list with all the applications existing in the project.

Link
Toolbar
Properties

Enter here the link which will open within the widget. It can be a Scriptcase application's name or an external URL. When you select an application, if it has links to other applications, you will be able also to set where these application links will opened.

Link

Calendar
 Chart
 Grid
 Control
 Search
 Form
 PDF Report

Close

Form

Form

Form

PDF Report

Toolbar

In the toolbar settings, it is possible to configure the buttons that will appear in the toolbar of the application that is configured in **connection**.

The toolbar options will be displayed according to the application being connected, that is, according to the toolbar of each application.

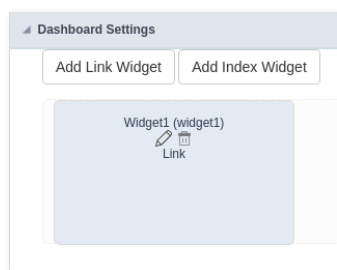
Link **Toolbar** Properties

Configure here the buttons that will be displayed in the application linked inside this widget. Note that if the button is not being originally used by the application, the flag here whether checked or not will not work.

- Column chart
- Line chart
- Area chart
- Pie chart
- Stacked chart
- Combination Chart
- Sorting
- Customize
- Chart types
- Summary
- PDF
- Print
- Dynamic search
- Advanced search

Add Index Widget

This option allows you to add to an Index Widget based on a database table.



Index

To create an Index Widget you need to make a few settings.

Index **Formatting** Filter Layout Properties

Configure the properties for retrieving the index value.

Title
Title1

Icon

Alignment
Center

Connection
conn_mysql

Table Name
orders

Metric field
priceorder

Metric function
Sum

Display as
Percentage difference

Period field
orderdate

Period function
Full date

Display values of
Both periods

Title

In this option we can define a title for the Widget for a better visualization of what we want to display.

Icon

In this option we can define an icon to be shown in the Widget.

Alignment

In this option we can define the horizontal position that the data in the Widget will be shown, we have three options:

Left: The left side of the Widget will be shown.

Centralized: Will be shown in the center of the Widget.

Right: This will be shown to the right of the Widget.

Connection

In this option it is allowed to choose which connection will be used in the Index Widget, each Widget can be configured with different connections.

Table Name

In this option it is allowed to choose which table will be used in the Index Widget.

After choosing the table, will be shown some more options, they are:

Metric Field

In this option it is allowed to select a table field to perform a metric, as in **chart application**. However it is possible to make some comparisons according to the period defined in the next option.

Period field

In this option it is allowed to select a field from the table so that the data chosen in the previous option can be compared.

Formatting

In this option you can configure the formatting of the metric and period that will be shown in the execution of the filter.

Metric Formatting

Metric Formatting as

Allows you to define the type of data that the metric will be shown, there are three options:

Internal: The index metric will be shown as a value of integer type. **Decimal:** The index metric will be shown as a value of type Decimal. **Currency:** The metric of the index will be shown as a value of type Modal.

Use regional settings

Lets you define whether the metric will follow the settings that were defined in the [Regional Settings](#).

Decimals

Allows you to set how many decimal places will be in the metric index, however this will only be available when the metric format is **Decimal** or **Currency**.

Complete with zeros

Allows you to set whether the value will be completed with zeros when set to **Decimal** or **Currency**.

Format the value in scales

Lets you set whether the number in the index will appear in a scale or the whole number. If **Yes** is chosen, the number instead of **10.000** will appear with **10 k** value.

Period Formatting

The options are shown according to the setting in **Feed Function** which is set to **Index**, so the setting can vary between:

Use regional settings

Allows you to set whether the period will follow the settings that were set in the [regional settings](#).

Display the name of the month

Lets you to set whether to display the name of the month.

Prefix for semester, four-month, quarter, two-month, week.

Lets you inform a `[lang][lang_sc]` or a title that the developer wants to appear.

Display name of the day

Lets you set whether to show the name of the day.

Filter

Allows you to perform the filter configuration that will be displayed when the Index Widget is running.

Configure the default filters of the index metric and dimension.

Metric filter SUM(priceorder)

Filter condition

Filter for the period YYYYMMDD2(orderdate)

Date range Relative period

Filter condition

Metric Filter

Filter conditions

In this option we can define the condition of the filter so that the information regarding that metric is shown, and it is not possible to change that at run time.

Period Filter

Date Ranges

In this option we can define the condition of the filter so that the information regarding that period is shown, and it is not possible to change at run time.

Relative period

Filter for the period

Date range Relative period

Select values

Year **Quarter** Month Week Day

Current year to date

Current year

Last year

Next year

We must define which periods are available for use in the filter.

Another option referring only to the relative period is the possibility of creating conditions with relative periods customized directly within the filter interface.

The available values are separated by tabs:

Year: Sets the year used in the filter. **Quarter:** Defines the quarters used in the filter. **Month:** Sets the usage of the months of the year. **Week:** Sets the use of the weeks of the year. **Day:** Sets the use of the days of the month.

Use field table filter

Allows you to create filters from table fields.

Use table's fields filter

#orderid

#customerid

#employeeid

orderdate

requireddate

shippeddate

#shipvia

Layout

In this option you can configure the layout of the map that will appear in the Widget.

Index Formatting Filter **Layout** Properties

Set up specifically for this widget some visual characteristics of its presentation. These values will overwrite those set in the application theme.

Index background

Title font

Positive index font

Negative index font

Neutral index font

Dimension font

Positive metric font

Neutral metric font

Index background

It allows you to set a background color for the widget content.

Title Source

Allows you to configure the font to be displayed in the index title, in this option there are four more options:

Arial, Helvetica, sans-serif; 12px; bold; #000000

Arial, Helvetica, sans-serif

12 #000000 bold

Family/source type Font size Color of the font The font style

Source of positive index

Allows you to configure the font that will be displayed in the index when positive, in this option there are four more options:

Arial, Helvetica, sans-serif; 12px; bold; #000000

Arial, Helvetica, sans-serif

12 #000000 bold

Family/source type Font size Color of the font The font style

Negative index source

Allows you to configure the font that will be displayed in the index when negative, in this option there are four more options:

Arial, Helvetica, sans-serif; 12px; bold; #000000

Arial, Helvetica, sans-serif

12 #000000 bold

Family/source type Font size Color of the font The font style

Source of the neutral index

Allows you to configure the font to be displayed in the index when neutral, in this option there are four more options:

Arial, Helvetica, sans-serif; 12px; bold; #000000

Arial, Helvetica, sans-serif

12 #000000 bold

Family/source type Font size Color of the font The font style

Source of the dimension index

Allows you to configure the font size, in this option there are four more options:

Arial, Helvetica, sans-serif; 12px; bold; #000000

Arial, Helvetica, sans-serif

12 #000000 bold

Family/source type Font size Color of the font The font style

Source of positive metrics

Allows you to configure the font that will be displayed in the metric when positive, in this option there are four more options:

Family/source type **Font size** **Color of the font** **The font style**

Negative metric source

Allows you to configure the font that will be displayed in the metric when negative, in this option there are four more options:

Family/source type **Font size** **Color of the font** **The font style**

Neutric metric source

Allows you to configure the font that will be displayed in the metric when neutral, in this option there are four more options:

Family/source type **Font size** **Color of the font** **The font style**

Properties

Allows you to configure some of the features that will be available at run time, these are:

Compact Mode

View Widget Header

Enables or disables the running widget header.

Remove

Enables or disables the button to delete the widget at run time.

Move

Enables or disables the option that allows widgets to be moved at runtime.

Maximize

Enables or disables the option to maximize the widget.

Expand

Enables or disables the button to expand the widget.

Recharge Time

Allows you to define in seconds a time for the Widget to be reloaded, thus reloading the information.

Security ?		
ATTRIBUTE	VALUE	DESCRIPTION
Use Security	<input type="checkbox"/>	Application needs validation to run. Read about the sc_apl_status macro.
Url output of the security	<input type="text"/>	Uri to be redirected if the user does not have access to this application.
Use Password	<input type="checkbox"/>	Enter password required to run the application.
Request password just once	<input type="checkbox"/>	Request password just once per session without asking for password every time you open the application in the same session.
Enable direct call by URL	<input checked="" type="checkbox"/>	Allow an application to be called by typing the URL directly in the browsers.

Application Security Configuration

Use Security

Turns the application accessible just when using the macro sc_apl_status or using the security module. When it is enabled, the application displays a message "User not Authorized" if the user tries to access it directly.

Url output of the security

It redirects the application to another URL after displays the warning "User not Authorized".

Use Password

Turns the application accessible just by using a password.

Request password just once

Request the password once per session, not needing to inform the password every time you access the application.

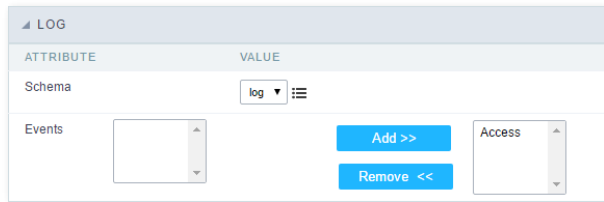
Enable direct call by URL

Allows to call the application directly from the URL in the browsers.

Allows calling the application directly from the URL in the browsers. In the development environment, you can disable those options for testing, it avoids requiring the login every time you run the application. To disable the security and password in development, you must access Options > My ScriptCase and uncheck the options: Enable Use of Security, Enable Use of Password.

This interface allows you to define a Log schema to the app. The Log scheme tracks what the end-users are doing in the app. Those are the events it can tracks: Access, Insert, Update, and delete.

To learn how to create a Log schema [see here](#).



The screenshot shows a configuration window titled "LOG". It features a table with two columns: "ATTRIBUTE" and "VALUE". The "Schema" attribute has a dropdown menu currently set to "log". Below the table, there are two empty list boxes: "Events" on the left and "Access" on the right. Between these two boxes are two blue buttons: "Add >>" and "Remove <<".

Application Log configuration

Schema Select an existing log schema for the application.

Events You can select just specific events to save in the log.

This module lets you editing display options for the application. You can apply visualization themes, organize blocks, define values, and the display format of the Header and Footer.

Layout Settings

ScriptCase creates an app by using default values defined per project, but you can change those values for a specific application on this interface.

LAYOUT SETTINGS

Header Template	<input type="text" value="Flat"/>	Template name used for the application header.
Footer Template	<input type="text" value="Default"/>	Template name used for the application footer.
Button	<input type="text"/>	Use different buttons than what was defined in the color scheme.
Themes	<input type="text" value="Sc9_Rhino"/>	Use different themes from the one defined for the color scheme

Header

|< << >> >| xxyyzz xxxxx yyyyy ▾

Block 1

Name

Type Male Female

Address*

Groups* Male Female

Countries ▾

Address

Photos

Drag & Drop files here

Captcha

Image1.png ✓

Image2.png ✗

Application Layout Settings

Header Template

Allows choosing the look of the application Header.

Footer Template

Allows choosing the look of the application Footer.

Button

Allows choosing the button theme for the application.

Themes

Choose one of the selected themes in the project properties. It defines the look of the application, like colors, fonts, and others.

Header & Footer

Header

Here we can define if we want to display the header or not, the application title, and fill the header variables. The header looks according to the selected template in the Layout settings interface.

HEADER	
ATTRIBUTE	VALUE
Display Header	<input checked="" type="checkbox"/>
Insertion Title	{lang_othr_frmj_tit} - orders
Update Title	{lang_othr_frmu_tit} - orders
Header Variables	
SC_HEADER	Title
SC_VALUE	Date

Application Header Configuration

Display Header


This option determines if we want to display the header or not.

Title

Allows informing the title of the application.

Header Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **itle**: It displays the value of "**Application Title**" in the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Depending on the Application, you may have more than one title option.

Footer

FOOTER	
ATTRIBUTE	VALUE
Display Footer	<input type="checkbox"/>
Footer Variables	
NM_FOOTER_DATA	
NM_FOOTER_LOGO	
NM_FOOTER_TITLE	

Application Footer Settings


This page may change depending on the footer template chosen in the Layout Settings.

Display Footer

This option determines if we want to display the footer or not.

Footer Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

General data

SETTINGS	
ATTRIBUTE	VALUE
Application Code	dashboard (9.00.0000)
Friendly URL	<input type="text"/>
Language	English (United States) ▼
Share Location Variable	<input checked="" type="checkbox"/>
Charset	<input type="text"/>
Share Theme Variable	<input checked="" type="checkbox"/>
Folder	root ▼
Edit by Project	<input checked="" type="checkbox"/>
Description	<input type="text"/>
HelpCase Link	<input type="text"/>

Settings interface.

In this folder you can set the application configuration attributes, such as:

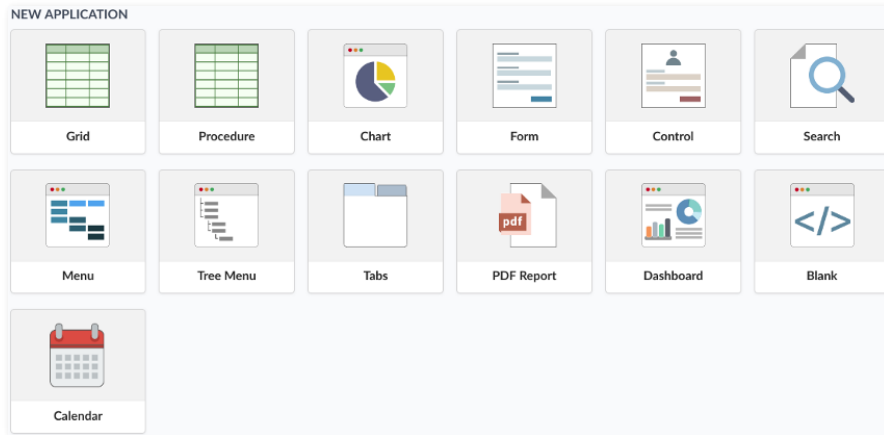
- **pplication Code** : Code for internal use of ScriptCase. Should start by a letter.
- **onnection** : Connection name of the used by ScriptCase to access the tables. The user can select it.
- **efault Profile** : Default profile to be loaded into the production environment.
- **anguage** : Data formatting based on the regional settings.
- **harsset** : Share location with other applications based on the values in the session.
- **hare Theme Variable** : Shares themes with other applications based on the values in the session.
- **xit URL** :URL to which the user will be redirected to exit the application.
- **older** : Folder where the application will be stored in the work group. The user can select it.
- **escription** : Application description.
- **elpCase Link** : Using this option you can associate HelpCase files to your application.

Global variables

New Application

The creation process of one application is simple enough and intuitive.

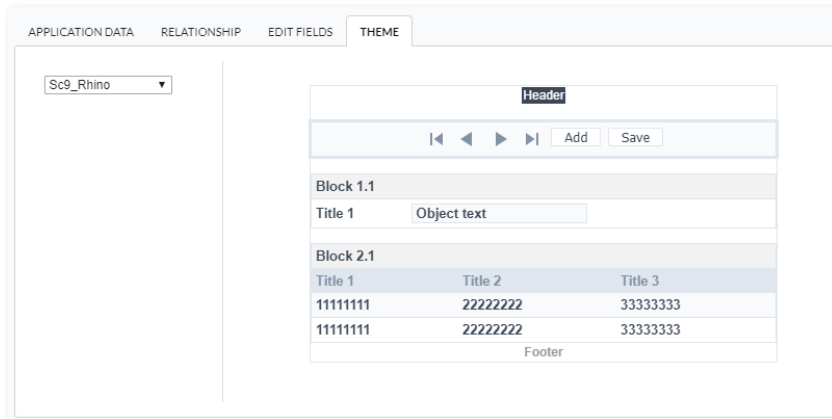
You just need to select which application you want to generate in the applications available in Scriptcase, That it will be generated automatically.



- **ame** - Application's name that is being created, It can not contain special characters.
- **ocalization** - Application's Language. The project's default language will be automatically selected.

Theme

In this screen you select the theme that the application will use. The default project's theme, that can be checked and changed in **Project > Properties**, will be selected automatically.



- **heme** - List of available themes in the project.

*After all the settings are done, click in the button **Create**. In the next screen the settings of the created application will be displayed.*

This interface is useful for editing the main display settings of the app.

SETTINGS	
ATTRIBUTE	VALUE
Friendly URL	<input type="text"/>
Line break in title	<input type="checkbox"/>
Horizontal Alignment	Center ▾
Margins	<input type="text"/> Up <input type="text"/> Down <input type="text"/> Right <input type="text"/> Left
Table Width	<input type="text" value="100"/>
Table Width Unit	Percent ▾
Table Columns	Automatic ▾

Form Settings Interface.

Friendly URL

Allows to define a URL for the application different from the application name. You can use the following chars (a-z, A-Z, 0-9, -). You still can change it on the initial screen of the home project, through the “Friendly URL” column in the apps list.

Line break in title

Use it if you want to break the line on the field titles.

Horizontal Alignment

Allows you to set the alignment of the application on the page.

Margins

Allows to define the margins of the application (Right, Left, Up and Down) in pixels.

Table Width

The width of the form table. Scriptcase uses Plain HTML to generate applications by using tables lines and cells.

Table Width Unit

Measurement unit for the table width defined in the previous option, being: percentage, pixel, or automatic.

Table Columns

This parameter defines the column (fields) width of the table (application).

Labels width

When the previous option is set to “Provided” you must inform the width of the labels here.

Layout and Behavior

This interface allows setting the behavior of the app.

LAYOUT AND BEHAVIOR	
ATTRIBUTE	VALUE
Automatic tab	<input type="checkbox"/>
Highlight Text on Focus	<input checked="" type="checkbox"/>
Use Enter to	▾
Field with Initial Focus	▾
Highlight Field with Error	<input type="checkbox"/>
Use a template from the HTML Editor	<input type="checkbox"/>

Layout and Behavior configuration Interface.

Automatic tab

Changes the focus to the next field when the amount of characters reaches its defined limit.

Highlight Text on Focus

Highlights the field when selected.

Use Enter to

Allows to use the “Enter” key to pass the focus to the next field.

Field with Initial Focus

Determines the field starts with focus when accessing the application. This option doesn't work with fields that contain a watermark.

Highlight Field with Error

Focus the field with the error when submitting the form.

Use a template from the HTML Editor

Allows to use the TinyMCE editor. You can edit and create your HTML Templates.

This interface is useful for editing the field settings and their position to display.

Edit Fields

1	Fields	Label	Datatype	New	Update	Read-only	Required	PK	DB value (Insert)	DB value (Update)
5	PAGE: PAG1		2	3		4				
6	BLOCK: FORM_CUSTOMERS									
	customerid	Customerid	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
	companyname	Companyname	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	contactname	Contactname	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	contacttitle	Contacttitle	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	birthdate	Birthdate	Date	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	country	Country	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
7	PAGE: FIELDS NOT SHOWN									
	regionid	Regionid	Integer	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	stateid	Stateid	Select	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	city	City	Text	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	address	Address	Text	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

Edit Fields Configuration.

Fields

It allows accessing the field settings (a pencil icon on the left). You can change the field position by dragging them to the desired position. Drag a field to "fields not displayed" if you don't want it in the app.

Label

It defines the title of the field in the app. For example: if the field name in the database is fld_txt_customer_name, you can display the label "Customer Name".

Data type

It informs the data type of the field.

New

It defines if the field is available when inserting new records.

Update

It defines if the field is available when updating records.

Read-Only

It defines the field as a label. The user can't change its value.

Required

It defines if the field must contain a value.

PK

It defines the Primary Keys fields.

DB value (Insert)

Defines a default value for the field when inserting a new record, like an auto-increment, Date, DateTime, or IP.

DB value (Update)

Defines a default value for the field when updating a record, like an auto-increment, Date, DateTime, or IP.

Page

It shows the pages available in the application. All apps have a page, at least. Each page contains one or more blocks.

Blocks

It shows the blocks available in the application. Blocks contain fields. All apps have a block, at least. A block is displayed if it contains one or more fields.

Page Fields Not Shown

Here we can see the fields that are not in the application.

Observe that you can drag any line to the desired position, blocks, and page. Pages contain Blocks, and Blocks contain fields.

Display Settings

Settings of messages display.

DISPLAY	
ATTRIBUTE	VALUE
Markers positioning	Right
Display message	<input checked="" type="checkbox"/>

Display Interface.

Attributes ### Markers positioning Set the position of the markers that indicates required fields.

Display message
Set it if you want to display the message of the required field.

This interface allows to sort the fields positioning.

Select the fields that will be displayed.

Available Fields	Selected Fields (Pag1)
orderid	form_orders
customerid	orderid
employeeid	customerid
orderdate	employeeid
requireddate	orderdate
shippeddate	requireddate
shipvia	
freight	
priceorder	
shipcountry	
shipregion	
shipstate	
shipcity	
shipname	
shipaddress	
shippostalcode	

Save Restore

Field Positioning

The left box has a list of all fields available in the application. Here you must select which fields you want to display in the app.

The right box contains the selected fields displayed in the application.

Use the arrows buttons between the boxes to select or deselect a field.

The application toolbar has two segments: Top and Bottom, in a way that is possible to define to display buttons into both areas. Those areas work independently, allowing them to display the same button, for example.

It's also possible to select the buttons and their position if the application is running on a mobile device.

Toolbar

Desktop

Here we must inform the toolbar settings for the "Classic Web Version" mode and which buttons are available in the application when accessed from a **Desktop** environment.

Mobile

Here we must inform the toolbar settings for the "Mobile Version" mode. That is which buttons are available in the application when accessed from a **Mobile** dispositive.

Toolbar

Configure the toolbar below for a "classic web version " and also for a "mobile version".

Desktop

Mobile

Top Toolbar

Others

- QuickSearch
- Dynamic Search
- Languages
- Themes
- HelpCase
- Rows Counter
- Jump to
- Copy

Navigation

- Navigation by page
- First
- Previous
- Next

Left

- QuickSearch
- Dynamic Search
- Center
- Insert
- Cancel
- Update
- Delete
- Right
- Exit

⏪ ⏩ ⏴ ⏵ ⏶ ⏷ ⏸ ⏹

Group

Add
Edit
Delete

Bottom Toolbar

Others

- QuickSearch
- Dynamic Search
- Languages
- Themes
- HelpCase
- Rows Counter
- Jump to
- Copy

Navigation

- Navigation by page
- First
- Previous
- Next

Left

- Jump to
- Center
- First
- Previous
- Navigation by page
- Next
- Last
- Right
- Rows Counter

⏪ ⏩ ⏴ ⏵ ⏶ ⏷ ⏸ ⏹

Group

Add
Edit
Delete

Navigation:

Buttons relative to the navigation of the application.

Next	Move to the next page that can be a single record or a list of records.
Previous	Returns to displays the previous page records or a single record.
First	Move to the First page or record
Last	Move to the Last page or record
Exit	Close the application
Navigation by page	Displays a "page-number" navigation bar. Example: 1 2 3 4 5

Export:

The options available to export the Records. Scriptcase generates the following export formats for Forms:

PDF	Generates all the data of the application in a PDF format.
Print	Creates an HTML with the records ready for printing.

Update:

The CRUD options available in the Form.

Insert	Inserts the record into the database.
Update	Saves the changes made in a record.
Delete	Deletes the selected record.
Cancel	Cancel the changes made in a record before the insertion.

Others:

Other options available in the Form application.

Jump To	Move to the informed page or record.
Copy	Copy the current record data to another one.
Quick Search	Perform a quick search in the records of the application.
Dynamic Search	It displays the fields of the search to filter the records.
Languages	Displays a Combobox with the languages available in the project properties.
Themes	Displays a Combobox with the languages available in the project properties.
Rows Counter	Displays the number of records retrieved in the application.
HelpCase	Displays a button to open the help page.

Separator:

-----	Displays a line separating the buttons.
-------	---

Toolbar Mobile

Toolbar

Configure the toolbar below for a "classic web version " and also for a "mobile version"

Desktop **Mobile**

Top Mobile toolbar

- Navigation
 - First
 - Previous
 - Next
 - Last
 - Exit
 - Navigation by page
- Export
 - PDF
 - WORD
 - XLS
 - XML
 - CSV
 - RTF

Left

- QuickSearch
- {lang_btns_expt_email}
- PDF (email)
- WORD (email)
- XLS (email)
- XML (email)
- CSV (email)
- RTF (email)
- {lang_btns_expt}
- PDF
-
- WORD
- XLS

Group

Add Edit Delete

Bottom Mobile toolbar

- Navigation
 - First
 - Previous
 - Next
 - Last
 - Exit
 - Navigation by page
- Export
 - PDF
 - WORD
 - XLS
 - XML
 - CSV
 - RTF

Left

- First
- Previous
- Rows Counter
- Next
- Last
- Center
- Right

Group

Add Edit Delete

Contains the same options as the **Desktop** version, adding only the item "Copy from Desktop", on which, when clicked, it copies the items from the **Desktop** toolbar to the **Mobile** toolbar.

Buttons Group

The **Group** option allows you to group a set of buttons of the application toolbar to display them as a dropdown, for example.

Toolbar

Configure the toolbar below for a "classic web version " and also for a "mobile version".

Desktop **Mobile** Copy from

Top Mobile toolbar

- Navigation
 - First
 - Previous
 - Next
 - Last
 - Exit
 - Navigation by page
- Export
 - PDF
 - WORD
 - Excel
 - XML
 - CSV
 - RTF

Left

- QuickSearch
- {lang_btns_expt}
- PDF
-
- WORD
- Excel
-
- XML
- CSV
- RTF
-
- Print
- {lang_btns_settings}

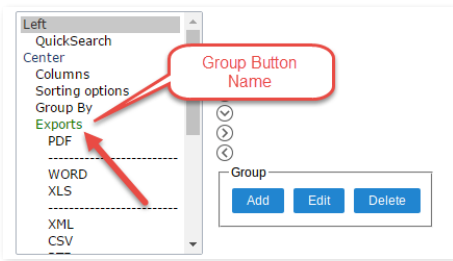
Group of buttons

Add Edit Delete

Add

Add a new group of buttons.

After creating a button group, you need to move the grouped buttons below of the Button Group and then move them to the right. Like the image below:



Buttons Settings

Button	Label	Hint	Shortcut key
Ok			
Exit			

Button:

It displays the buttons available in the application.

Label:

Allows defining the buttons name that to display for the users.

Hint:

Allows defining the buttons hint that to display for the users.

Shortcut Key:

Allows defining the shortcut key for each button.

Application Hotkeys

Scriptcase allows creating shortcut keys to your applications. You can select a predefined template or create specific actions for an application.

VALUE	DESCRIPTION				
<input checked="" type="checkbox"/> toggleHotkeys();	Define if the application will use hotkeys				
SC_DefaultHotkeys	Select the hotkey template from previously created schemas				
Clear <input type="button" value="+"/>					
<table border="1"> <thead> <tr> <th>ACTION</th> <th>KEYBINDING</th> </tr> </thead> <tbody> <tr> <td colspan="2" style="text-align: center;">No hotkeys configured</td> </tr> </tbody> </table>		ACTION	KEYBINDING	No hotkeys configured	
ACTION	KEYBINDING				
No hotkeys configured					
Clear <input type="button" value="+"/>					

Use hotkeys

Defines if the application uses hotkeys. When you enable this option, the default shortcut keys settings are disabled.

Hotkeys template

Select the [hotkey template](#) previously created.

Action

Selects the triggered action when pressing the selected key.

Keybinding

Selects the keys responsible for executing the chosen action.

Add "+"

Adds a new action on the keys list.

Clear

It clears the selected hotkeys preference.

Options

ATTRIBUTE	VALUE
Format Row Counter	<input type="text" value="{lang_othr_smry_info}"/>
The number of links displayed	<input type="text" value="5"/>
Jump to	<input type="text" value="Page"/>
Records per page	<input type="text" value="10,20,50"/>
Help by Block	<input type="checkbox"/>
General Help	<input type="checkbox"/>
Toolbar buttons	<input type="text" value="A DIV below the toolbar."/>

Rows Counter:

Allows to display the Rows Counter or not.

Example: (1 to 10 of 200)

The Number of Links Displayed:

Defines the number of links per page, when the navigation option is disabled.

Jump To:

Move to the informed record page.

Records by page:

Allows defining the number of records to be exhibited for each page. To show all records use the option "all".

Example: 10,20,30, all

Help by Block:

Indicates if the helps messages from relatives fields are grouped by block. In each field, we can define a help text. With this option activated, it shows up an icon in the block title bar to call the help page.

General Help:

The General Help "consolidates" all the fields help pages in a single page, putting an icon in the toolbar to call the help page.

Toolbar Buttons:

The type of view for the Toolbar buttons (A DIV below the toolbar, Modal).

This interface allows configuring the related database settings, such as the Primary Key, Filters, Sorts.

SQL SETTINGS	
ATTRIBUTE	VALUE
Select primary key fields.	
	<div style="display: flex; align-items: center;"> <div style="margin-right: 10px;"> <input type="button" value="On/Off"/> <input type="button" value="All"/> <input type="button" value="None"/> </div> <div style="border: 1px solid #ccc; padding: 5px;"> <ul style="list-style-type: none"> *orderid customerid employeeid orderdate requireddate shippeddate shipvia freight priceorder shipcountry </div> </div>
Where clause	<input type="text"/>
Order By	<input type="text"/>
Connection	<input type="text" value="conn_mysql"/>
Table Name	<input type="text" value="orders"/>
Variable for Table	<input type="text"/> <input type="text"/>
Case Sensitive	<input checked="" type="checkbox"/>

SQL configuration Interface.

Select primary key fields

It lets you define the Primary key of the Form. ScriptCase already identifies Primary Keys, but you can manually inform or change it by using the buttons beside the fields list. See how the buttons work:

- **n/Off** : Adds or Removes the attribute that defines the primary key for the field. The primary key fields have an asterisk beside their names.
- **ll** : Defines all fields as Primary Keys.
- **one** : Defines none fields as Primary Keys.
- **orting Button** : These are the arrows on the right side of the Combo box. It allows ordering the fields of the Primary Key, placing it in the desired order. To order them, click on the field and use the arrows to move it.

Where clause

It allows adding a WHERE clause to filter the SQL records.

Order By

It allows adding an ORDER BY clause to determine the order to display the records. By default, it uses the primary key to sort the records.

Connection

It allows defining the database connection of the application. You can change the connection to another one that has the same table.

Table Name

It informs the database table used in the Form.

Variable for Table

It allows to use a variable to change a part of the string containing the table name.

Var	<input type="text"/>
orders	<input type="text"/>

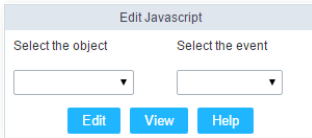
Variable for Table Configuration.

Fill the first input with the name of the variable (replace to). The second input you should fill with the part of the name of the table you want to replace (replace from).

Case sensitive

It defines if the database connection uses case sensitive or not.

To coding with JavaScript in Scriptcase, we must associate a JavaScript event to a form field.



Edit JavaScript Interface.

Select the object

This Combo box displays the fields of the form application, and also the form itself as an object

Select the event

Use it to associate the event with the selected field to apply the JS code. View the events available:

- **nClick**: Acts when clicking on the field.
- **nBlur**: Acts when the focus leaves the object.
- **nChange**: Acts when the focus leaves the object, and there are changes in the value.
- **nFocus**: Runs when the focus gets in the object.
- **nMouseOver**: Runs when the mouse pointer hovers the object.
- **nMouseOut**: Runs when the mouse pointer moves out the object.

Events related to the Form

The events below are associated directly with the form object.

- **nLoad**: This event runs when the page is loaded, also when clicking on the navigating buttons.
- **nSubmit**: This event runs when clicking on the “New”, “Save”, and “Delete” buttons.

Edit JavaScript

*Select the object and event, then click on the Edit button. It opens a page to inform custom JavaScript routines and

```
![Edit JavaScript Interface][javascript_edicao_code]
*Edit JavaScript Interface*
```

> The JavaScript language doesn't have the same behavior in all the browsers available. A tip is to test running the

- **nClick Example**
 - hen clicking on a field of type radio, you can enable or disable form fields according to the selected value.

```
if(document.F1.gender[0].checked){
  document.F1.maternity.disabled = false;
  document.F1.maternity.style.background='FFFFFF'
}
if(document.F1.gender[1].checked){
  document.F1.maternity.disabled = true;
  document.F1.maternity.style.background='FCEEB3'
}
```

To access the values of a radio field, you need to use the index.

- **nBlur Example**
 - ou can define a warning for the field “weekly_work_time” when the focus is getting out it.

```
if (document.F1.tp_point[0].checked && document.F1.weekly_work_time.value > '20')
{
  alert("The work time exceeds the limit allowed");
  document.F1.weekly_work_time.value = "";
  document.F1.weekly_work_time.focus();
}
```

- **nChange Example**
 - y modifying the “Salary” of an employee and leaving the field, we’ll check if his “position” is “gardener”.

```
if (document.F1.salary.value > 2000.00 && document.F1.position.value == 'gardener'){
  alert('When I grow up, I want to be a gardener);
}
```

- **nFocus Example**
 - fter informing the purchase value and selecting the payment method in a Select object “Select: pay_method”, the JavaScript code below calculates the value of the purchase.

```
if (document.F1.pay_method[document.F1.pay_method.selectedIndex].text == 'Money')
{
  document.F1.total.value = document.F1.paurchase_value.value;
}
```

- **nMouseOver Example**

- You can change the style (background color, font and font color) when the mouse hovers the field.

```
document.F1.field_name.style.backgroundColor = "0FFCCA"
```

- **nMouseOut Example**

- Resets the background color when the mouse moves out from the field.

```
document.F1.field_name.style.backgroundColor = "FFFFFF"
```

Security		
ATTRIBUTE	VALUE	DESCRIPTION
Use Security	<input type="checkbox"/>	Application needs validation to run. Read about the sc_apl_status macro.
Url output of the security	<input type="text"/>	Uri to be redirected if the user does not have access to this application.
Use Password	<input type="checkbox"/>	Enter password required to run the application.
Request password just once	<input type="checkbox"/>	Request password just once per session without asking for password every time you open the application in the same session.
Enable direct call by URL	<input checked="" type="checkbox"/>	Allow an application to be called by typing the URL directly in the browsers.

Application Security Configuration

Use Security

Turns the application accessible just when using the macro sc_apl_status or using the security module. When it is enabled, the application displays a message "User not Authorized" if the user tries to access it directly.

Url output of the security

It redirects the application to another URL after displays the warning "User not Authorized".

Use Password

Turns the application accessible just by using a password.

Request password just once

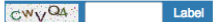
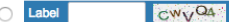
Request the password once per session, not needing to inform the password every time you access the application.

Enable direct call by URL

Allows to call the application directly from the URL in the browsers.

Allows calling the application directly from the URL in the browsers. In the development environment, you can disable those options for testing, it avoids requiring the login every time you run the application. To disable the security and password in development, you must access Options > My ScriptCase and uncheck the options: Enable Use of Security, Enable Use of Password.

Captcha

Captcha		
ATTRIBUTE	VALUE	DESCRIPTION
Use Captcha	<input type="checkbox"/> Captcha	Enables Captcha function for this application.
Number of Characters	<input type="text" value="4"/>	Number of characters
Character List	<input type="text" value="ABCDEFGHIJKLMNOPQRSTUVWXYZ01234"/>	Character list.
Label	<input type="text" value="{lang_othr_cptc_lbel}"/>	Message shown near Captcha.
Error message	<input type="text" value="{lang_othr_cptc_errm}"/>	Captcha error message.
Height	<input type="text" value="65"/>	Picture height in pixels.
Width	<input type="text" value="175"/>	Picture width in pixels.
Font Size	<input type="text" value="20"/>	Font size of the captcha.
Reload	<input type="checkbox"/>	Add a button to reload Captcha.
Select one layout		
<input checked="" type="radio"/>  <input type="radio"/> 		

Use Captcha

Defines if the application uses Captcha.

Number of Characters

Amount of characters in the Captcha image.

Character List

List of characters used in the Captcha.

Label

The message displayed for the Captcha.

Error message

Captcha error message.

Height

Height of the image generated by the Captcha (in pixels).

Width

Width of the image generated by the Captcha (in pixels).

Font Size

Size of the font used in the image generated by the Captcha (in pixels).

Reload

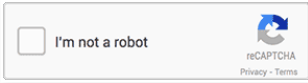
Display the refresh button in the Captcha.

Select one layout

It offers various layouts for the Captcha display.

reCAPTCHA

reCAPTCHA is an API provided by Google for forms. It adds security, preventing automatic submission of forms through robots.

reCAPTCHA sample:

1. first, we must request an API Key to activate reCAPTCHA into a Scriptcase application by following the steps below:

To get a **Site key** and **Secret Key** go to the link: <https://www.google.com/recaptcha/admin#list>. See the image:

Register a new site

Label

Choose the type of reCAPTCHA ?

reCAPTCHA V2
 Validate users with the "I'm not a robot" checkbox.

Invisible reCAPTCHA
 Validate users in the background.

reCAPTCHA Android
 Validate users in your android app.

Domains
(one per line)

Accept the reCAPTCHA Terms of Service.
By accessing or using the reCAPTCHA APIs, you agree to the Google APIs Terms of Use, and to the Additional Terms below. Please read and understand all applicable terms and policies before accessing the APIs.
» reCAPTCHA Terms of Service

Send alerts to owners ? **Register**

Label

It is a project identifier to create the reCAPTCHA keys.

Choose the type of reCaptcha

We must choose the option **reCAPTCHA V2**.

Domains

We can insert multiple domains (one per line) to limit the API uses.

1. then, we need to accept the Terms of Service ("Accept the reCAPTCHA Terms of Service").
2. then clicking on **Register**, the page refreshes and shows the integration of reCAPTCHA information. There we can get the **Site Key** and **Secret Key**:

Adding reCAPTCHA to your site

Keys

<p>Site key <small>Use this in the HTML code your site serves to users.</small></p> <input type="text"/>	<p>Secret key <small>Use this for communication between your site and Google. Be sure to keep it a secret.</small></p> <input type="text"/>
--	---

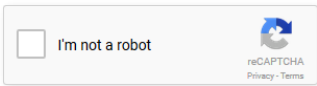
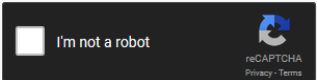
1. now, we can set the Scriptcase application security:

ATTRIBUTE	VALUE
Use Captcha	Recaptcha
Site Key	
Secret Key	
Theme	light
Type	Audio
Size	Compact
Position	Center

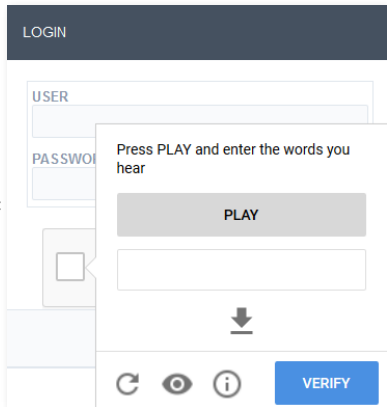
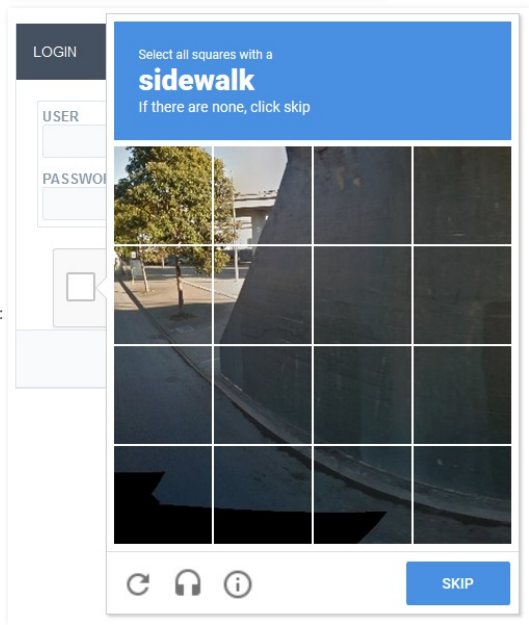
Site Key
Key generated by Google after reCAPTCHA project creation.

Secret Key
Key generated by Google after reCAPTCHA project creation.

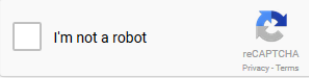
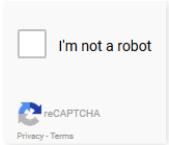
Theme
Defines the reCAPTCHA color. There are two options:

- **light** : 
- **dark** : 

Type
The type of reCAPTCHA. There are two options:

- **audio**: 
- **image**: 

Size
The size of reCAPTCHA. There are two options:

- **normal**:  A rectangular widget with a light gray background. On the left, there is a small square checkbox followed by the text "I'm not a robot". On the right, there is a circular reCAPTCHA logo with the text "reCAPTCHA" and "Privacy - Terms" below it.
- **compact**:  A rectangular widget with a light gray background. On the left, there is a small square checkbox followed by the text "I'm not a robot". On the right, there is a circular reCAPTCHA logo with the text "reCAPTCHA" and "Privacy - Terms" below it.

Position

Here we can define the reCAPTCHA component alignment:

- **eft**: Position the reCAPTCHA component at the left.
- **enter**: Position the reCAPTCHA component at the center.
- **ight**: Position the reCAPTCHA component at the right.

This interface allows you to define a Log schema to the app. The Log schema tracks what the end-users are doing in the app. Those are the events it can tracks: Access, Insert, Update, and delete.

To learn how to create a Log schema [see here](#).

The screenshot shows a configuration window titled "LOG". It features a table with two columns: "ATTRIBUTE" and "VALUE". The "Schema" attribute has a dropdown menu currently set to "log". Below the table, there are two empty list boxes: "Events" on the left and "Access" on the right. Between these two boxes are two blue buttons: "Add >>" and "Remove <<".

Application Log configuration

Schema Select an existing log schema for the application.

Events You can select just specific events to save in the log.

Scriptcase can link its applications using OAuth methods. These methods allow you to integrate a Scriptcase project with the following data sources:

Facebook

FACEBOOK	
ATTRIBUTE	VALUE
App ID	<input type="text"/>
Secret	<input type="text"/>
Method return	<input type="text"/> +

Facebook fields authentication interface.

When you create a button Facebook authentication, you must fill in the fields:

- **pp ID** : Facebook application ID. This ID will be provided by [Facebook Developer's page](#).
- **ecret** : Secret key provided by Facebook, so the application can be executed.
- **ethod return** : Method that will be performed when Facebook service returns the requested data. You will have to create that method according to our validation rules, using the option "Programming > PHP Methods"

For building applications, visit: [Facebook app development](#)

Twitter

TWITTER	
ATTRIBUTE	VALUE
Key	<input type="text"/>
Secret	<input type="text"/>
Method return	<input type="text"/> +

Twitter fields authentication interface.

When you create a button "Twitter Authentication", you must fill in the fields:

- **ey** : Twitter application ID. This ID will be provided by [Twitter Developer's page](#).
- **ecret** : Secret key provided by Twitter, so the application can be executed.
- **ethod return** : Method that will be performed when Twitter service returns the requested data. You will have to create that method according to our validation rules, using the option "Programming > PHP Methods"

For building applications, visit: [Twitter app development](#)

Google

GOOGLE	
ATTRIBUTE	VALUE
Application Name	<input type="text"/>
Client ID	<input type="text"/>
Secret	<input type="text"/>
Method return	<input type="text"/> +

Google fields authentication interface.

- **pplication name** : Name of the application created on Google+.
- **lient ID** : Google+ application ID. This ID will be provides by [Google Developer's page](#).
- **ecret** : Secret key provided by Google+, so the application can be executed.
- **ethod return** : Method that will be performed when Google+ service returns the requested data. You will have to create that method according to our validation rules, using the option "Programming > PHP Methods"

For building applications, visit: [Google+ app development](#)

PayPal

PAYPAL	
ATTRIBUTE	VALUE
Paypal URL	<input type="text" value="https://www.sandbox.paypal.com/cgi-bin/webscr"/>
Id	<input type="text"/>
Order number	<input type="text"/>
Customized field	<input type="text"/>
Variable with the description	<input type="text"/>
Variable with the Total amount	<input type="text"/>
Variable with the currency type	<input type="text"/>
Paypal Method	<input type="text" value="v"/> +
Method Cancel	<input type="text" value="v"/> +
Method IPN	<input type="text" value="v"/> +

PayPal fields authentication interface.

To create the integration with PayPal, you must fill in these fields:

- **ayPal URL** : This URL can be to the PayPal SandBox or to the real PayPal API. The SandBox is used to perform tests with the application before applying the integration in a real environment.
- **d** : PayPal ID to receive the sales money. Usually the registered email as business.
- **rder number** : Order Number to be stored in PayPal. Also used to treat the receipt of the IPN in order to identify and process the order as paid.
- **ustomized field** : Custom field to be stored in PayPal. Also used to treat the receipt of the IPN in order to identify and process the order as paid, if you want to do some additional checking.
- **ariable with the description** : Using this field you can place the Global variable that will contain the description to be sent to PayPal.
- **ariable with the Total amount** : Method to be ran when PayPal service return the data requested.
- **ariable with the currency type** : Use to specify the payment currency. If the currency variable is not included, the currency defaults to USD.
- **ayPal Method** : Method that will be called after the User complete the PayPal transaction. You will have to create that method according to our validation rules, using the option "Programming > PHP Methods"
- **ethod Cancel** : Method to be executed if the user cancels the PayPal transaction. You will have to create that method according to our validation rules, using the option "Programming > PHP Methods"
- **ethod IPN** : Method that will be called by web service from PayPal to confirm the payment of a transaction. This method is called by web service and it has no HTML output to the end-user. You will have to create that method according to our validation rules, using the option "Programming > PHP Methods"

For PayPal configuration, please visit: [Configuring PayPal credentials](#)

The variables list returned by the PayPal API can be found below:

```
[mc_gross]
[invoice]
[protection_eligibility]
[address_status]
[payer_id]
[tax]
[address_street]
[payment_date]
[payment_status]
[charset]
[address_zip]
[first_name]
[mc_fee]
[address_country_code]
[address_name]
[notify_version]
[custom]
[payer_status]
[business]
[address_country]
[address_city]
[quantity]
[verify_sign]
[payer_email]
[txn_id]
[payment_type]
[last_name]
[address_state]
[receiver_email]
[payment_fee]
[receiver_id]
[txn_type]
[item_name]
[mc_currency]
[item_number]
```

[residence_country]
[test_ipn]
[handling_amount]
[transaction_subject]
[payment_gross]
[shipping]
[ipn_track_id]

1. [Home](#)
2. Applications

General Overview

The applications can have two types of fields.

Table Fields

Columns of the table connected to the application and their data types.

Example:

In a table with the following structure:

```
CREATE TABLE `orders` (
  `orderid` int(11) NOT NULL,
  `customerid` varchar(5) NULL DEFAULT NULL,
  `employeeid` int(11) NULL DEFAULT NULL,
  `orderdate` date NULL DEFAULT NULL,
  `requireddate` date NULL DEFAULT NULL,
  PRIMARY KEY (`orderid`) USING BTREE
);
```

These are the listed fields on left menu **Fields**:

Virtual Fields

Fields created inside application to assist in the development process.

How to create a virtual field

1. Inside an application, access the left menu **Fields** and click on **New Field** option.
2. Choose how many fields do you want to create:

1. Create the fields and define type, name and label for each one:

- **Type:** Data type of the field.
- **Name:** Internal name of the field. Used to identify the field on events and Scriptcase interface.
- **Label:** Title of the field displayed on the application.

Scriptcase **doesn't insert** virtual fields on the table after its creation.

Types of fields by applications

Application Table Fields Virtual Fields

Grid	<input type="text"/>	<input type="text"/>
Procedure	<input type="text"/>	<input type="text"/>
Chart	<input type="text"/>	<input type="text"/>
Form	<input type="text"/>	<input type="text"/>
Control	<input type="text"/>	<input type="text"/>
Search	<input type="text"/>	<input type="text"/>
PDF Report	<input type="text"/>	<input type="text"/>
Calendar	<input type="text"/>	<input type="text"/>

The applications **Menu**, **Tree Menu**, **Dashboard** and **Blank** were not listed because they haven't manage fields.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Control
- 4. Text

Text

General Settings

This type of field allows the developer to create quickly inputs to insert and update data, where the final user can inform its data to be allocated in its database.

Type of Data

Defines the type of the field in the application. In this case we should select **Text**.

Label

Defines the title that will be displayed in the field when the application is executed. The terminology used in the interface is fundamental to the system have a good usability, who should use names and familiar terms to the final user of the application, instead of using terms from the system.

For example, this text field that has the name **customerid**, the client would have a much better understanding of the functionality of the field when we define the label as **Customer Name**.

Not only a fixed text, the **Label** attribute allows the use of langs to define the field's title, making it possible to use your application in a multi language project.

Watermark

Informing a text to the **Watermark** it will be displayed in the input a text as an example that can be informed in the field. The result after set will be this:

Initial Value(type)

Allows the initial definition to the field when the form is in insert mode. It is possible to choose between two options:

Defined Value: When this option is selected, the Initial Value attribute will be available, where we should inform the field's initial value. For example, my initial value is **Arlindo**, when a new register is inserted, the field **Seller Name** will be initialized as Arlindo.

System Data: When this option is selected, the initial value will be the actual date of your computer's system.

Amount of Characters

Allows to set the width of the text field's input that varies with the amount of characters informed. Although, if the amount of characters typed are greater than the setting, the text will be pushed to the left, to keep the maximum amount of characters as defined.

Show HTML Content

When this option is active every HTML, CSS and JavaScript content that are in the database will be displayed with the main value.

Validation Image

When this option is active, a image will be displayed next to the informed field if the field is according to the settings defined by the developer.

In the example below, the field was set to receive at least 5 characters, see what happens when informed only 4 characters:

However if informed 5 or more characters the field will be displayed as:

Password Field

When this option is active, the text field will be converted to the format used in password fields. For example:

Save Variable

Allows to save a session variable(global variable) with the field's value, to be used in others applications.

For example, in the login form the username can be saved in session and displayed on the header of others applications.

Variable Name

In this attribute we should define the name of the session variable, active in the previous item, that will receive the field's value.

We should inform only the variable's name, - **var_rating**.

The method to use its value is [global variable](#).

Field Mask

Defines the field mask. There are two types of mask described in the table below:

Character Description

- | | |
|---|--|
| X | It is overridden by any characters returned by the database. Filled is required and the value will be completed with leading zeros when there are fewer characters than the size of the mask. |
| Z | It is overridden by any characters returned by the database. Its filling is optional and when there are fewer characters than the mask size, nothing will be done about the characters that are missing. In addition the leading zeros will be suppressed. When used in conjunction with the X necessarily the use of this character is to the left of the mask. |
| 9 | It represents any numeric character (from 0-9) |
| A | It represents an alpha numeric character (A-Z,a-z) |
| * | It represents any alpha-numeric character (A-Z,a-z,0-9)entered by the user. |

For example, it is possible to set the mast to display a telephone number:

It will be show with this format on runtime:

It is also possible to set the field mask like those examples:

Field mask examples:

Telephone number

Field	Mask Input	Typed Value	Formatted Value
Phone Number	+99 99 9999 - 9999	123456789012	+12 34 5678 - 9012
Phone Number	(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
Phone Number	(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
Phone Number	(zz) xxxx - xxxx	12345678	() 1234 - 5678
Software Key (Letters only)	AAAA-AAAA-AAAA-AAAA	QWERTYUIASDFGHJK	QWER-TYUI-ASDF-GHJK
Software Key (Letters and numbers)	xxxx-xxxx-xxxx-xxxx	Q1W2E3R4T5Y6U7I8	Q1W2-E3R4-T5Y6-U7I8
License Plate	AAA - 9999	QWE1234	QWE - 1234
Scriptcase Serial Key	A999A999A999-*	D111H111G111DG2P	D111H111G111-DG2P
Multiple masks (Phone Number)	9999-9999;(99)9999-9999; 9999 999 9999		+99 99 9999-9999

Validate with mask

When this option is active it is possible to insert the data with the mask set in the **Field Mask** option.

Field size in database

Defines the field's size related to the size set in the database. This value is already set automatically by default when the application is generated.

Hidden Field

When this option is active, the field will be hidden in the application on runtime.

Label Field

When this option is active, the field will be altered to only a label where the info will be displayed, where updates or inserts will not be possible.

Disable Field

Allows the developer to disable the field, so the user can not type a value according with the option defined by the developer.

The available options are: **No** - Initial value of the attribute, this option does not disable the field.

Update Mode - This option only disables the field when editing existing registers.

Insert Mode - This option only disables the field when inserting new registers.

Insert / Update Mode - This option disables the field when editing existing registers or when inserting new registers.

HTML Type

HTML object used to display the field in the form.

SQL Type

Informs the type of the field in the database.

Values Format

Format of Values Interface

- **Case Settings** : Allows to convert the letters of the field when losing the focus. The options are:
 - **Upper Case** : All in Upper Case.
 - **Lower Case** : All in Lower Case.
 - **Capitalize first word** : Capitalizes the first letter of the word.
 - **Capitalize all words** : Capitalizes the first letter of all the words.
- **Minimum Size** : Allows to define the minimum amount of characters accepted for the field.
- **Maximum Size** : Allows to define the maximum amount of characters accepted for the field.
- **Allowed Characters** : Allows to select a set of characters that can be typed into the field. The options are:
 - **All** : Allows any type of character to be typed into the field
 - **Selected** : Defines a set of characters allowed. Using the configuration below, will be allowed to type letters, numbers and other characters (defined in the attribute More). The letters a, b and c (defined in the Less Attribute) are not allowed.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Control
- 4. Multiple Lines Text

Multiple Lines Text

General Settings

Multiple Lines Text field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Multiple Lines Text , you can inform a Text value to the field in multiple lines.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Lines** : Allows to define the amount of lines that the field will have at start.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Field size in database** : Determines the field size in bytes. It is used to determine the max size of characters that is allowed to type in.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface

- **Case Settings** : Allows to convert the letters of the field when losing the focus. The options are:
 - **Upper Case** : All in Upper Case.
 - **Lower Case** : All in Lower Case.
 - **Capitalize first word** : Capitalizes the first letter of the word.
 - **Capitalize all words** : Capitalizes the first letter of all the words.
- **Minimum Size** : Allows to define the minimum amount of characters accepted for the field.
- **Maximum Size** : Allows to define the maximum amount of characters accepted for the field.
- **Allowed Characters** : Allows to select a set of characters that can be typed into the field. The options are:
 - **All** : Allows any type of character to be typed into the field
 - **Selected** : Defines a set of characters allowed. Using the configuration below, will be allowed to type letters, numbers and other characters (defined in the attribute More). The letters a, b and c (defined in the Less Attribute) are not allowed.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.

- **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
- **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Control
- 4. Integer

Integer

General Settings

Data type:

It sets the application field type. When the field is set as Number, it is permissible to define formatting rules of integers.

Label:

It sets the title that will be displayed in the field when you run the application. The terminology used in the interface is fundamental for your system has a good usability. In this case, we recommend you to use names and terms familiar to the end user of the application, instead of using terms originated in the system.

Watermark

Informing a text for the **watermark** it will be displayed on the input text with an example of what can be reported in the field. The result after that will be configured:

Initial value (type)

Allows setting an initial value for the field when the application is in inclusion. You can choose between two options:

Pre-set value: When you select this option, the Initial Value attribute will be made available in the place where we inform the initial value of the field. For example, my initial value is **Arlindo** when inserting a new record the **seller_name** field will start with Arlindo.

System date: When you select this option, the initial value will be the current system date of your computer.

If you select the type system date, it will be not necessary to fill in the initial value attribute.

Amount of characters

It allows you to set the width of the input text field according to the amount of characters. However, if the quantity entered is greater than the set for the characters, the text will be pushed to the left, in order to ensure the maximum amount of characters set in the option of **Values formatting**.

Validation Image

When you enable this option, an image will be shown next to the field informing whether the field is in accordance with the settings of *minimum size* and *Maximum size* (as you can see in the images below) defined in the Formatting value option.

In the example below, the field was set to receive at least 5 characters, see what happens with the image when it receives only 4 characters:

However, if the value inserted has 5 or more characters the the image will change according to example bellow:

Use slider:

It displays a slider component in the field. So you can increase or decrease the value sliding the cursor. You can also customize the increment value, if it increments the value 1 by 1, 2, 5, 10... N.

Field Mask Input:

It sets the mask input for the field. There are two types of mask as described in the table below:

Character Description

X	It is overridden by any characters returned by the database. Filled is required and the value will be completed with leading zeros when there are fewer characters than the size of the mask.
Z	It is overridden by any characters returned by the database. Its filling is optional and when there are fewer characters than the mask size, nothing will be done about the characters that are missing. In addition the leading zeros will be suppressed. When used in conjunction with the X necessarily the use of this character is to the left of the mask.
9	It represents any numeric character (from 0-9)
A	It represents an alpha numeric character (A-Z,a-z)
*	It represents any alpha-numeric character (A-Z,a-z,0-9)entered by the user.

Mask Input Examples:

Field	Mask Input	Typed Value	Formatted Value
Phone Number	+99 99 9999 - 9999	123456789012	+12 34 5678 - 9012
Phone Number	(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
Phone Number	(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
Phone Number	(zz) xxxx - xxxx	12345678	() 1234 - 5678
Software Key (Letters only)	AAAA-AAAA-AAAA-AAAA	QWERTYUIASDFGHJK	QWER-TYUI-ASDF-GHJK
Software Key (Letters and numbers)	xxxx-xxxx-xxxx-xxxx	Q1W2E3R4T5Y6U7I8	Q1W2-E3R4-T5Y6-U7I8
License Plate	AAA - 9999	QWE1234	QWE - 1234
Scriptcase Serial Key	A999A999A999-**	D111H111G111DG2P	D111H111G111-DG2P
Multiple masks (Phone Number)	9999-9999;(99)9999-9999; 9999 999 9999		+99 99 9999-9999

Validate with Mask

By enabling this option you will be able to enter data according to the mask that was configured in the option **Field Mask** and Scriptcase will validate it.

Record Variable

Allows you to record a session variable with the value of the field ([global variable] [var_glob]), to be used in other applications.

Example, in the login form the username can be saved in session and displayed in the header of other applications.

Variable Name

In this option you must set the session variable name, enabled in the previous item, which will receive the value of the field.

You need to inform only the variable name, for example: **var_seller**.

The recovery of the value is made as [Global Variable][var_glob]{:target='blank'}.

Field size in Database

This option sets the size of the field relative to the size that is configured in the database. By default this value is already configured automatically when the application is generated.

Hidden Field

This option when enabled will hide the field inside the application at the time of execution.

Label Field

By enabling this option, the field will be changed to only one label where the information will be displayed, so it is not possible to make changes or inserts in the field configured as label.

Disable field

Allows you to disable the field, making it impossible for the user to enter a value according to the developer-defined option.

The available options are:

No - initial value of the attribute, this option does not disable the field. **Update mode** - This option disables the field only when editing the records. **Insert mode** - This option disables the field only inserting new records. **Update/Insert mode** - This option disables the field in both editing and inserting new records.

HTML Type

HTML object used to display the field in the application.

Tipo SQL:

Reports the SQL type of the field as it is configured in the database.

Values Format

Use regional settings:

It allows you to apply regional settings to the number fields formatting. When this option is not selected it will be displayed the attributes group, minus sign and negative number format.

Digit grouping:

It allows you to define if the field displays the digit grouping separator.

Maximum Size:

It allows you to define the maximum size that will be used in the field of the application.

Minimum Value:

It allows you to define the minimum value that will be used in the field of the application.

Maximum Value:

It allows you to define the maximum value that will be used in the field of the application.

Grouping:

It allows you to define which character is used to separate thousands. This option is only available when the option to use regional settings is disabled.

Negative sign:

It allows you to define which character will be used for the display of negative numbers. This option is only available when the option to use regional settings is disabled.

Negative number format:

It allows you to define the placement of the negative sign in relation to value. This option is only available when the option to use regional settings is disabled.

Accept:

It allows to define if the field accept only negative values, positive-only or both.

Display Calculator:

This option when enabled will display a calculator to assist the end user to perform calculations.

Use Spin:

This option applies a Spin component to the number field. So the end user can increase or decrease the numbers using it. You will also be able to set a minor or major range for the spin.

Smaller increment for the Spin::

Sets the minimum range of the Spin component.

Greater increment for the Spin:

Sets the maximum range of the Spin.

Lookup Settings

The Lookup is a way used to provide the end user a list of values description/name where the items can be selected and used in the application. For example, for a field sex that in the database is represented with M or F values can be presented as male or female dynamically with the lookup. So you will be able to retrieve these values from a database table or manually, if the reference table does not exist.

Example:

Lookup Methods

There are two lookup methods available:

Manual: To use this method it is necessary to inform the selection conditions manually to the Lookup.

Automatic: To use this method it is necessary to use a lookup query from the database to access the reference table.

Editing-automatic lookup:

Select Command:

It sets the SQL command that will retrieve the value to display inside the Grid field. The command must have the following format

```
SELECT field_to_be_displayed FROM table WHERE key_field = {grid_field}
```

The Grid field must be always referenced between curly braces {}, So at run time, the field braces will be replaced by its value in the table.

Multiple Options:

When the Select command informed return multiple values, the option must be selected.

Delimiter:

It sets the tab of values returned by the Select command, this option should be filled when the "Yes" option is checked in the multiple attribute options.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

Use a specific connection:

This option allows you to select another existing connection in the project, other than the current connection, to select the values for the lookup.

Edit Lookup - Manual:

Lookup used to modify the display of the field with predefined values.

Lookup Types

It sets the operating mode of the lookup can be: single value, multiple values (delimiter), multiple values (position) and multiple values (binary).

Single value: it will be displayed only one value for the field. There is a direct replacement of the value stored in the Bank for a label defined in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value of "M" will be replaced by "Male".

Multiple Values (Delimiter):

By using this option it will be displayed multiple values for the selected field. The values must be separated by a delimiter to be informed. A replacement of parts from a string stored in the field, separated by a delimiter for values contained in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value S;M will be replaced by Sports and Movies.

Delimiter: Character used to separate the values within the same string.

Multiple Values (Position)

Allows you to recover information stored in a single string of the selected field. In order for this information to be retrieved must be informed, in addition to the label, the starting position and the number of bytes that each information occupies within the string.

As an example we will use a string to store Sex, Marital status and Hobby respectively. Sex occupies one byte, Status one byte and Hobby two bytes .

For this purpose we define the list as:

Label	Value	Start	Size
Male	M	1	1
Female	F	1	1
Married	M	2	1
Single	S	2	1
Study	ST	3	2
Sportes	SP	3	2
Reading	LE	3	2

Example: the string MMST would be presented in the query as: Male Married Study

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value M will be replaced by Male.

Start: : Starting position in the string where the information is recorded. The first position is always 1.

Size: : Number of bytes that the information occupies in the string.

Multiple values (binary):

It allows you to retrieve several information stored in decimal form of the selected field.

As examples we will use the following list (although not informed in the interface of inclusion of values in the list, each item has a value assigned automatically according to the order):

Assigned value	Description in Lookup
1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

To display the data, a decomposition of the decimal number stored in the bank is performed. For example, numbers 11 and 12 (written in the database) would be decomposed as follows:

$$11 = 1 + 2 + 8 = (\text{Sports} - \text{Culture} - \text{Reading})$$

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

$$12 = 4 + 8 = (\text{Leisure} - \text{Reading})$$

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

Label:

Text that will be displayed in the Grid.

Include Button:

Includes in the list the values populated in the Label and value fields.

Alter Button:

It changes the attributes of the selected item.

Delete Button:

Deletes the selected item.

Clean Button:

It cleans the fields

Save Button:

Allows the user to save the entire list, for later use using the Load Lookup definition option.

Load Lookup definition:

Allows you to load a predefined value list for use as a lookup.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**

- **Font** : Allows to choose the font type, that will be applied to the application field title.
- **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
- **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.

- **CSS of the Field**

- **Font** : Allows to choose the font type, that will be applied to the application field.
- **Font Size** : Allows to choose the the font size, that will be applied to the application field.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
- **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the field.
- **Height** : To define a height for the field.

- **CSS of the Input Object**

- **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
- **Font Size** : Allows to choose the the font size, that will be applied to the application field.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the Input Object.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
- **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the Input Object.
- **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.**Help Description**

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Control
- 4. Decimal

Decimal

General Settings

Decimal field configuration Interface.

- **Data Type** : Select the type of field for the application. When set to Decimal, you can inform decimal values to the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Defines the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Defines the amount of characters allowed for the field.
- **Validation Image** : Displays an image next to the field when it's being validated.
- **Save Variable** : Saves a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Defines the name for the session variable that'll receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Applies the Regional Settings to format the number of the field. When not enabled, the attributes Digit Grouping Separator, Decimal Separator, Negative sign and Negative number format.
- **Digit grouping** : Defines if the field will display the digits separator.
- **Maximum Size** : Determines the max size of the field.
- **Minimum Value** : Set the min value of the field.
- **Maximum Value** : Determines the max value of the field.
- **Decimal Precision** : Defines the amount of decimal places for the field.
- **Digit Grouping Separator** : Defines the character that is used to separate the thousand.
- **Decimal Separator** : Defines the decimal separator.
- **Negative sign** : Defines the character that will be used to display negative numbers.
- **Negative number format** : Defines the position of the Negative sign em relation to the value.
- **Accept** : Determines if the field, will accept only negative, positive or both numbers.
- **Show a calculator** : Displays an icon next to the field, which allows the user to calculate a value for the field.
- **Auto-complete with zeros** : If enabled, it automatically informs the decimal places when not specified. If disabled, the user needs to always inform the decimal value.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.** : When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.

- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
- **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Control
- 4. Currency

Currency

General Settings

Currency field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Currency, you can currency values to the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through Javascript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the datatype of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the number of the field. When not enabled, the attributes Digit Grouping Separator, Decimal Separator, Negative sign and Negative number format.
- **Digit grouping** : Defines if the field will display the digits separator.
- **Currency symbol usage** : Defines if the field will display the Currency Symbol of the Regional Settings.
- **Maximum Size** : Determines the max size of the field.
- **Minimum Value** : Determines the min value of the field.
- **Maximum Value** : Determines the max value of the field.
- **Decimal Precision** : Defines the amount of decimal places for the field.
- **Digit Grouping Separator** : Defines the character that is used to separate the thousand.
- **Decimal Separator** : Defines the decimal separator.
- **Negative sign** : Defines the character that will be used to display negative numbers.
- **Negative number format** : Defines the position of the Negative sign em relation to the value.
- **Accept** : Determines if the field, will accept only negative, positive or both numbers.
- **Show a calculator** : Displays an icon next to the field, which aids the user to calculate a value for the field.
- **Auto-complete with zeros** : If enabled, it automatically informs the decimal places when not specified. If disabled, the user needs to always inform the decimal value.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.** : When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.

- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
- **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Control
4. HTML Image

HTML Image

General Settings

HTML Image field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as HTML Image, it allows to place an image to the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Image** : Defines an image that will be displayed. The icon "Select Image", lists all the standard images from scriptcase and also the image that you have uploaded to scriptcase. The icon "Upload an image" allows the developer to send an image to the server which is from another machine.
- **Border** : Border size for the image in Pixels.
- **Width** : Define the width of the image in Pixels.
- **Height** : Define the height of the image in Pixels.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Control
4. Label

Label

General Settings

Label field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as label, the field only display the output of the value .
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Text** : You can inform the text that will display beside the field.
- **Reload** : When enabled, make the text informed for the field to become a link to reload the form when clicked.
- **Position** : Allows to position the field in the chosen block.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Control
4. Card

Card

General Settings

Credit Card Number Configuration Interface.

- **Data Type** : Defines the type of field for the application. When it is defined as a Credit Card Number, the field verifies if the value is valid.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name would be "Client Name".
- **Watermark** : Creates a placeholder on the field with the text informed.
- **Initial Value (type)** : Set the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Defines the amount of characters allowed for the field.
- **Validation Image** : Display an image next to the field when it is being validated.
- **Save Variable** : Saves a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Set the name for the session variable that will receive the field value.
- **Field size in database** : Determines the size in bytes of the fields. It is used to determine the max amount of characters to be typed in.
- **Field Mask** : Defines the display mask for the field. There are three types of masks that can be merged.

Character	Description
9	Represents a numeric character (0-9)
A	Represents an alpha numeric character (A-Z,a-z)
*	Represents any alpha numeric character (A-Z,a-z,0-9) typed by the user.

It is possible to merge two or more masks simultaneously, separated by a semi coma with the smallest mask at start. The replacement occurs when the user is typing when the lowest amount of character exceeds.

Examples of Masks

Field	Mask	Informed Value	Formatted Value
Telephone Number	+99 99 9999 - 9999	123456789012	+12 34 5678 - 9012
Software Key (Only Numbers)	AAAA-AAAA-AAAA-AAAA	QWERTYUIASDFGHJK	QWER-TYUI-ASDF-GHJK
Software Key (Letters an Numbers)	**_**_**	Q1W2E3R4T5Y6U7I8	Q1W2-E3R4-T5Y6-U7I8
License Plates	AAA - 9999	QWE1234	QWE - 1234
ScriptCase License	A999A999A999.**	D111H111G111DG2P	D111H111G111-DG2P
Multiple Masks (Telephone)	9999-9999;(99)9999-9999; 9999 999 9999		+99 99 9999-9999

- **Hidden Field** : This option makes the field hidden, but still allows it is value to be processed through JavaScript or PHP.
- **Label Field** : This option make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Display the data type of field in the database.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Control
4. Card type

Card type

General Settings

Credit Card field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Credit Card , it'll setup a combo-box for you to select a creditcard type.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it'll be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that'll receive the field value.
- **Field size in database** : Determines the field size in bytes. It's used to determine the max size of characters that's allowed to type in.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that'll be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface.

- **Credit Card** : Allows to define which cards will be listed and their internal value.
- **Use Title/Deny** : Allows to inform a new value if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen. The label and the value of this option are defined as the Negative Value and the Title.
- **Negative value** : Value of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Title** : Label of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Related Field** : You'll inform which of the fields contains the values of a Credit Card Number so that the validations can be done correctly.
- **Reload Form** : Reloads the form when the field's value is modified.

Lookups Settings

Method used to place a description next to the field.

- **SQL lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.

- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
- **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the field.
- **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Control
- 4. E-mail

E-mail

General Settings

Email field configuration Interface.

- **Data Type** : Defines the type of field for the application. When set to Email, the field applies validations for an email format.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Defines the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Defines the amount of characters allowed for the field.
- **Validation Image** : Displays an image next to the field when it is being validated.
- **Save Variable** : Saves a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Defines the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through Javascript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Display Icon** : Displays the Email icon next to the field.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface.

- **Credit Card** : Allows to define which cards will be listed and their internal value.
- **Use Title/Deny** : Allows to inform a new value if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen. The label and the value of this option are defined as the Negative Value and the Title.
- **Negative value** : Value of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Title** : Label of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Related Field** : You'll inform which of the fields contains the values of a Credit Card Number so that the validations can be done correctly.
- **Reload Form** : Reloads the form when the field's value is modified.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** : Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).

- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Control
4. URL

URL

General Settings

URL field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to URL, you can inform a Link value to the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Display Icon** : Displays an icon beside the field.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **Show the URL as a clickable link** : Sets the content of the field to clickable link.
- **Target handling where the link will open** : Target where the the lick will direct to after being clicked
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface.

- **Credit Card** : Allows to define which cards will be listed and their internal value.
- **Use Title/Deny** : Allows to inform a new value if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen. The label and the value of this option are defined as the Negative Value and the Title.
- **Negative value** : Value of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Title** : Label of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Related Field** : You'll inform which of the fields contains the values of a Credit Card Number so that the validations can be done correctly.
- **Reload Form** : Reloads the form when the field's value is modified.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.

- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
- **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Control
4. HTML Color

HTML Color


General Settings

 HTML Color field configuration Interface.


- **Data Type** : Select the type of field for the application. When set to HTML Color, you can select a color to be used in the form.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Initial Value (type)** : Set the initial value to the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

 Initial Value Configuration Interface.

- **Amount of characters** : Defines the amount of characters allowed for the field.
- **Validation Image** : Displays an image next to the field when it is being validated.
- **Save Variable** : Save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Defines the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

 Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of the field in the database.

Values Format

 Format of Values Interface.

- **Credit Card** : Allows to define which cards will be listed and their internal value.
- **Use Title/Deny** : Allows to inform a new value if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen. The label and the value of this option are defined as the Negative Value and the Title.
- **Negative value** : Value of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Title** : Label of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Related Field** : You'll inform which of the fields contains the values of a Credit Card Number so that the validations can be done correctly.
- **Reload Form** : Reloads the form when the field's value is modified.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.

- **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:


Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Control
4. HTML Editor


HTML Editor

General Settings

 HTML Editor field configuration Interface.

- **Data Type** : Defines the type of field for the application. When set to HTML Editor, you can inform any type of character and they will be saved in HTML form.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Height** : Set the Height in pixels to the HTML Editor field.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through Javascript or PHP.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **SQL Type** : Displays the data type of field in the database.


Toolbar

 HTML Editor toolbar settings Interface.


- **Properties**
 - **Position** : Location of the HTML Editor toolbar.
 - **Alignment** : Toolbar button alignment.
 - **Status** : Status Bar Display (Do not Display, Top and Bottom).
 - **Amount** : The Amount of lines of the HTML Editor toolbar.
- **Button Organization** : Positions the toolbar buttons.
- **Preview** : You can visualize the toolbar according to the previous settings.

Database Value


Allows to define a value to be saved to the database.

 Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

 Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

 Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** : Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** : Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** : Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.

- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the Input Object.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
- **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the Input Object.
- **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Control
4. Localization

Localization

General Settings

Location field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it's defined as Location, you will see a combobox with the Languages that are part of the Project Properties.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Reload** : When enabled, make the text informed for the field to become a link to reload the form when clicked.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Control
4. Theme

Theme

General Settings

Theme field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Theme, you view a combo box with the list of themes that are set to your project.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Reload** : Reload the form after modifying the value.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

-
- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
 - **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
 - **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Control
4. Date

Date

General Settings

Date field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Date, you can inform a date.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows it is value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator attribute.
- **Date separator** : Allows you to inform the separator symbol for the date.
- **First Day** : Define the first day of the week.
- **Display** : Select the format of the day for display.
- **Internal Format** : Allows to define the format the field when the SQL type is different from **Date**. You need to use the characters **Y**, **M** and **D** that correspond to **Year**, **Month** and **Day**.
- **Use Combo-box** : Allows you to select the date using a combo-box.
- **Minimum Date** :
 - **Fixed Date** : Inform the minimum date allowed presented in the selected date format.
 - **Current Date** : Clicking on the icon next to the text field, ScriptCase offers the following options:
 - **Actual date** : It will place the current date as the max allowed date.
 - **Actual date with increment** : The minimum date will be the current date (+) the days or months or years that you want to increment.
 - **Actual date with decrement** : The minimum date will be the current date (-) the days or months or years that you want to decrement.
- **Maximum Date** :
 - **Fixed Date** : Inform the minimum date allowed presented in the selected date format.
 - **Current Date** : Clicking on the icon next to the text field, ScriptCase offers the following options:
 - **Actual date** : It will place the current date as the max allowed date.
 - **Actual date with increment** : The minimum date will be the current date (+) the days or months or years that you want to increment.
 - **Actual date with decrement** : The minimum date will be the current date (-) the days or months or years that you want to decrement.
- **Display Format** : Enables the Date format beside the field when informing the date.
- **Display Calendar** : Enables the a calendar icon beside the field, this allows to select the date from a calendar with the format already setup.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.** : When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Control
- 4. Time

Time

General Settings

Time field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Time, you can inform a time to this field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator and time separator attributes.
- **Time separator** : Allows you to inform the separator symbol for the time.
- **Display** : Select the format of the time for display.
- **Internal Format** : Allows to define the format the field when the SQL type is different from **TIME**. You need to use the characters **HH**, **II**, and **SS** that correspond to **Day**, **Hour**, **Minutes** and **Seconds**.
- **Display Format** : Enables the Time format beside the field when informing the time.
- **Use Time picker** : Use the JQuery plug-in to choose the time.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets `{}`. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** : Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).

- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Control
- 4. Date and time

Date and time

General Settings

Date and Time field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Datetime, you can inform a date and time to this field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows it is value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator and time separator attributes.
- **Date separator** : Allows you to inform the separator symbol for the date.
- **Time separator** : Allows you to inform the separator symbol for the time.
- **Display** : Select the format of the day/time for display.
- **Internal Format** : Allows to define the format the field when the SQL type is different from **DATETIME**. You need to use the characters **Y, M, D, HH, II, and SS** that correspond to **Year, Month, Day, Hour, Minutes** and **Seconds**.
- **Display Format** : Enables the Date and Time format beside the field when informing the date.
- **Group Date and Time** : Groups the date and time in the same field.
- **Display Calendar** : Enables the a calendar icon beside the field, this allows to select the date from a calendar with the format already setup.
- **New Calendar** : Show the new jquery calendar, or the old calendar.
- **Years Limit** : The amount of years that will display in the calendar.
- **View week number** : Display the week number in the calendar.
- **Additional months** : Display additional months in the calendar.
- **Show Combo year and month** : Allows you to select the year and month using a combo-box.
- **Decimal of seconds** : Defines the amount of decimals for the seconds.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.

- **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
- **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Control
4. Image (File Name)

Image (File Name)

General Settings

Image (File Name) field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as an Image (File Name), all the images files are stored and loaded in a directory of the server (Only the image name is stored in the database).
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Image Border** : Width of the Image border. Value in Pixels.
- **Image Height** : Image height size. Value in Pixels.
- **Image Width** : Image width size. Value in Pixels.
- **Maintain Aspect** : Maintains the original aspect ratio of the image when resizing it.
- **Progress bar** : Displays a progress bar when sending the files to the server.
- **Upload area** : Displays a drag and drop area to upload the file.
- **Increment file** : Increments the file name if the theirs an existing one with the same name in the upload folder.
- **Sub-folder** : Sub-folder's name that the files are stored. The is sub folder is relative to the directory defined for the Document upload (see the Settings). It is possible to use global variables or local variables to format the name of the sub-folder.
- **Create Sub-folder** : When enabled, it creates the sub-folder if not already created.
- **Image Caching** : Time in minutes that the image cache will be stored.
- **Hide image name** : When enabled, only the image will be displayed without it is name.
- **Files Deletion** : Deletes the files from the directory when the record is deleted from the database.
- **Display link only** : WWhen enabled, you will be presented with a link to open the image in another image with the original size of the image (without using the re-dimensioning).
- **Open in Another Window** : Allows to open the image in another window.
- **Extensions to allow** : Extensions allowed to upload, separated by a semi coma (;). If left in blank, all the extensions are allowed. e.g. docx;jpg/png/xls
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Control
4. Document (File Name)

Document (File Name)

General Settings

This option allows to upload multiple files and store them on a table referencing a record from another table.

To use this option you first need to create a virtual field.

Multi-Upload Configuration.

- **Amount of columns** : Amount of columns to displays the files in the Field.
- **Files sorting** : Enables files sorting after the upload.
- **Position of the delete** : Checkbox option to delete files position.
- **Position on the upload** : Position of the records when uploading.
- **Show status** : Display the upload status.
- **Table** : Select the table that will store the information of the files when uploading.

After selecting the table, you will see a list of all the fields of the chosen table, where you will associate the parameters for each field when inserting a new record or updating an existing record.

Multi-Upload fields of the table Configuration.

- **File Name** : Obtains the name of the file that is being loaded.
- **Upload** : Obtains the binary value of the file.
- **Auto Increment** : Used only when field in the database in an auto increment or similar.
- **Manual Increment** : The form will be responsible for managing the auto-increment, calculating the value of the field before inserting the record.
- **Foreign Key** : Associates the file with a value of a field in the current form.
- **Defined** : You can specify a constant value or use a global variable. [global_var]
- **Date of Insertion** : Obtains the data of the server when inserting the record.
- **Datetime of Insertion** : Obtains the data and time of the server when inserting the record.
- **User IP** : The field will contain the IP of the computer that is accessing the application.
- **Calculated by the Database** : The value of the field is applied in the database.
- **Calculated by the Database if empty** : The value of the field is applied in the database if not informed by the application.
- **Null** : The field will the null value.

Watch bellow a video about Multiupload option:



Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).

- **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the Input Object.
- **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Control
- 4. Select

Select

General Settings

Select field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Select, you can select multiple option from a combo box (Select Field).
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.
- **Use Select2** : Uses the new component for data selection, allowing searches within the select.
- **Display Select2 search area** : Sets whether to display the search field within Select2.

Initial Value Configuration Interface.

- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Field size in database** : Determines the field size in bytes. It is used to determine the max size of characters that is allowed to type in.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through Javascript or PHP.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

• Lookup Methods

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

Selecting the lookup type.

◦ Lookup Method - Automatic

Lookup used to list the values that will be displayed on the Checkbox field. These values will be recovered dynamically using the Select command.

Automatic Lookup Interface.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimitation.

- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter than (;).
- **Height** : Defines the height for the select object.
- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

- **Link** : Allows to create a link to another form allowing to manipulate the list displayed on the select field. After the manipulation, the select object it updated automatically.

◦ Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).

- **Single Value** :

Setting up Single Value Lookup

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Default** : With this option enabled, the selected item will be inserted to the table when committed.
- **Height** : Defines the height for the select object.

- **Multiple Values (delimiter)**

You can store various values for the select field. The values are separated by the informed delimiter. For example: the combination **Sport, Cinema and Tourism** selected in form will be stored like **E;C;T** in case the delimiter is ; (semi coma).

Setting up Multiple Values (delimiter)

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Height** : Defines the height for the select object.

- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man, Single and Read**, in the database would be stored the following value **MSRD**.

Setting up Multiple Values (position)

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Negative** : Value to be saved in the table in case there is not any valued selected on the field.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.
- **Height** : Defines the height for the select object.

- **Multiple Values (binary)**

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute Value	Lookup Description
1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports, Pleasure and Reading** were selected, the number stored in the table would be **13**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) =13

Editing Lookup Configuration Multiple Values (Binary)

- **Label** : Text that will be displayed in the item list of the select field.
- **Height** : Defines the height for the select object.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

Displaying the original lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Lookup Method - Actual value

This lookup is used to list all the values in the selected field.

This lookup will apply a "distinct" to your SQL query.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Control
4. Double Select

Double Select

General Settings

Double Select field configuration Interface.

- **Data Type** : Select the type of field for the application. When set to Double Select you can have multiple options selected.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Initial Value (type)** : Defines the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Save Variable** : Saves a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Set the name for the session variable that will receive the field value.
- **Field size in database** : Determines the size in bytes of the fields. It is used to set the max amount of characters to be typed in.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through Javascript or PHP.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Displays the data type of field in the database.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or F can be presented like Male or Female or in a dynamic way, getting these values from the database.

Lookup Settings Display for the field. Automatic Lookup Interface..

- **SQL Select Statement** : Defines the SQL command that will get the values displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.

- **Height** : Set the height(lines) of the field interface.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter than (;).
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Selects another specific connection existing in the project. The select command will be done on the second connection.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).

- **Width** : To define a width for the field.
- **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Control
- 4. CheckBox

CheckBox

General Settings

CheckBox field configuration Interface.

- **Data Type** : Select the type of field for the application. When set to CheckBox, you can have multiple options selected.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Initial Value (type)** : Defines the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Save Variable** : Saves a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Define the name for the session variable that will receive the field value.
- **Field size in database** : Determines the size in bytes of the fields. It is used to determine the max amount of characters to be typed in.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through JavaScript or PHP. .
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or F can be presented like Male or Female or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

• Lookup Methods

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

Selecting the lookup type.

◦ Lookup Method - Automatic

Lookup used to list the values that will be displayed on the CheckBox field. These values will be recovered dynamically using the Select command.

Automatic Lookup Interface.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.

- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Option check all** : Displays two options on the field to check and uncheck all.

- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Selects another specific connection existing in the project. The select command will be done on the second connection.

◦ Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

Manual Lookup Interface.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
 - **Single Value** :

Setting up Single Value Lookup

- **Label** : Text that will be displayed in the item list of the checkBox.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Default** : With this option enabled, the selected item will be inserted to the table when committed.
- **Columns** : Set the amount of columns, for the list of items.

- **Multiple Values (delimiter)**

You can store various values for the checkBox field. The values are separated by the informed delimiter. For example: the combination **Sport, Cinema** and

Tourism selected in form will be stored like **E;C;T** in case the delimiter is ; (semi coma).

Setting up Multiple Values (delimiter)

- **Label** : Text that will be displayed in the item list of the checkBox.
 - **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
 - **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
 - **Columns** : Set amount of columns, for the list of items.
- **Multiple Values (position)**

Stores a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Martial Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in the database would be stored the following value **MSRD**.

Setting up Multiple Values (position)

- **Label** : Text that will be displayed in the item list of the checkBox.
 - **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
 - **Negative** : Value to be saved in the table in case there is not any valued selected on the field.
 - **Start** : Starting position of the string that is going to be stored. The first position is always 1.
 - **Size** : Amount of bytes that is going to occupy in the string.
 - **Columns** : Set the amount of columns, for the list of items.
- **Multiple Values (binary)**

Stores a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

Setting up Multiple Values (Binary)

- **Label** : Text that will be displayed in the item list of the checkBox.
- **Columns** : Allows you to inform the amount of columns, for the list of items.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Saves all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Refreshes the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Option check all** : Displays two options on the field to check and uncheck all.

- **Display original and lookup value**: Displays the value of the field and the value returned from the lookup.

- **Separated by**: Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

• CSS of the Title

- **Font**: Allows to choose the font type, that will be applied to the application field title.
- **Font Size**: Allows to choose the the font size, that will be applied to the application field title.
- **Font Color**: Allows to choose a color for the font from the color pallet.
- **Background Color**: Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold**: Applies the bold style to the font.
- **Underline**: Applies the underline style to the font.
- **Border Size**: Applies the border size to the title of the field.
- **Border Color**: Allows to choose a color for the border, using a color pallet to apply to the title.
- **Horizontal Alignment**: Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment**: Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width**: To define a width for the title of the field.
- **Height**: To define a height for the title of the field.

• CSS of the Field

- **Font**: Allows to choose the font type, that will be applied to the application field.
- **Font Size**: Allows to choose the the font size, that will be applied to the application field.
- **Font Color**: Allows to choose a color for the font from the color pallet.
- **Background Color**: Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold**: Applies the bold style to the font.
- **Underline**: Applies the underline style to the font.
- **Border Size**: Applies the border size to the field.
- **Border Color**: Allows to choose a color for the border, using a color pallet to apply to the field.
- **Horizontal Alignment**: Allows to position the field in the desired position (left,right,center and justify).
- **Vertical Alignment**: Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width**: To define a width for the field.
- **Height**: To define a height for the field.

• CSS of the Input Object

- **Font**: Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
- **Font Size**: Allows to choose the the font size, that will be applied to the application field.
- **Font Color**: Allows to choose a color for the font from the color pallet.
- **Background Color**: Allows to define the color for the Input Object, the color can be selected from the color pallet.
- **Bold**: Applies the bold style to the font.
- **Underline**: Applies the underline style to the font.
- **Border Size**: Applies the border size to the Input Object.
- **Border Color**: Allows to choose a color for the border, using a color pallet to apply to the Input Object.
- **Horizontal Alignment**: Allows to position the Input Object in the desired position (left,right,center and justify).
- **Vertical Alignment**: Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width**: To define a width for the Input Object.
- **Height**: To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Control
4. Radio

Radio

General Settings

Radio field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Radio, your allowed to select one of the options listed.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Field size in database** : Determines the size in bytes of the fields. It is used to determine the max amount of characters to be typed in.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through Javascript or PHP. .
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

• Lookup Methods

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

Selecting the lookup type.

◦ Lookup Method - Automatic

Lookup used to list the values that will be displayed on the Checkbox field. These values will be recovered dynamically using the Select command.

Automatic Lookup Interface.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.

- **Columns** : Allows you to inform the amount of columns, for the list of items.

- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.

- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.

- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

◦ Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

Manual Lookup Interface.

- **Label** : Text that will be displayed in the item list of the radio.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Default** : With this option enabled, the selected item will be inserted to the table when committed.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones

saved by the user.

- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

Displaying the original lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Ajax Settings

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

• CSS of the Title

- **Font** : Allows to choose the font type, that will be applied to the application field title.
- **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
- **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.

• CSS of the Field

- **Font** : Allows to choose the font type, that will be applied to the application field.
- **Font Size** : Allows to choose the the font size, that will be applied to the application field.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
- **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the field.
- **Height** : To define a height for the field.

• CSS of the Input Object

- **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
- **Font Size** : Allows to choose the the font size, that will be applied to the application field.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the Input Object.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
- **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the Input Object.
- **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Control
- 4. Profile

Profile

General Settings

Profile field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Profile, it will display a combo with the profiles setup in the project properties.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Control
4. Menu Links

Menu Links

General Settings

Menu Links field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as Menu Links, you can define links to be listed in the application.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through Javascript or PHP.
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Values Format Interface.

- **Menu Items** : Control the Insert, Update and Delete the items of the menu.
 - **Label** : Defines the label for the menu item.
 - **Link** : Defines the external link that'll be called when clicking the item.
 - **Icon** : Defines the icons that'll be displayed beside the link.
 - **Insert** : Insert a new item to the menu.
 - **Update** : Update the selected item.
 - **Delete** : Delete the selected item.
 - **Clear** : Clears the data of the selected item.
 - **Clear All** : Removes all items form the menu.
- **Display Mode** : Defines how the menu items will display, having the types: Select , Radio and Link:
 - **Select**
 - **Radio**
 - **Link**
- **Hide Label** : When using the display mode, you can enable or disable the label display.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Control
4. Text Auto-Complete

Text Auto-Complete

General Settings

Text auto complete field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Text auto complete, you can inform a value based on the select statement of the lookup settings and it will manipulate an internal Text for the data.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You'll inform the Defined Value here.
- **Use Select2**: Uses the new component for data selection, allowing searches within the select.
- **Amount of characters for the Select2**: Sets the number of characters to start the search in Select2.
- **Amount of lines for the Select2**: Sets the maximum number of rows to list the search result in Select2.
- **Width for the Select2**: Sets a width for the area for the Select2.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Complete to the Left** : Allows to define the character that will be used to complete the value to the left that the user typed in to the max size of the field defined in the Field size in database option.
- **Field size in database** : Determines the field size in bytes. it is used to determine the max size of characters that is allowed to type in.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Validate on submit** : Validate the field only when the form is submitted.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface

- **Case Settings** : Allows to convert the letters of the field when losing the focus. The options are:
 - **Upper Case** : All in Upper Case.
 - **Lower Case** : All in Lower Case.
 - **Capitalize first word** : Capitalizes the first letter of the word.
 - **Capitalize all words** : Capitalizes the first letter of all the words.
- **Minimum Size** : Allows to define the minimum amount of characters accepted for the field.
- **Maximum Size** : Allows to define the maximum amount of characters accepted for the field.
- **Allowed Characters** : Allows to select a set of characters that can be typed into the field. The options are:
 - **All** : Allows any type of character to be typed into the field
 - **Selected** : Defines a set of characters allowed. Using the configuration below, will be allowed to type letters, numbers and other characters (defined in the attribute More). The letters a, b and c (defined in the Less Attribute) are not allowed.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field.
- **Rows** : Allows to define the amount of records displayed in the field.
- **Width** : Defines the width the size of capture box (Capture Text).
- **Search options** : Allows to define the search settings of the field(Start equal to, Any part and End equals to).
- **Capture Text** : When not enabled, you will only view the field to inform the data. See an example below. *lookup Settings configuration Interface.*
- **Display original and lookup value**: Displays the value of the field and the value returned from the lookup.

lookup Settings validation configuration Interface.

- **Show label with the description** : Displays a label with the description.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Control
4. Number Auto-Complete

Number Auto-Complete

General Settings

Number auto complete field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Number auto complete, you can inform a value based on the select statement of the lookup settings and it will manipulate an internal number for the data.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Complete to the Left** : Allows to define the character that will be used to complete the value to the left that the user typed in to the max size of the field defined in the Field size in database option.
- **Field size in database** : Determines the field size in bytes. It is used to determine the max size of characters that is allowed to type in.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Validate on submit** : Validate the field only when the form is submitted.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface

- **Case Settings** : Allows to convert the letters of the field when losing the focus. The options are:
 - **Upper Case** : All in Upper Case.
 - **Lower Case** : All in Lower Case.
 - **Capitalize first word** : Capitalizes the first letter of the word.
 - **Capitalize all words** : Capitalizes the first letter of all the words.
- **Minimum Size** : Allows to define the minimum amount of characters accepted for the field.
- **Maximum Size** : Allows to define the maximum amount of characters accepted for the field.
- **Allowed Characters** : Allows to select a set of characters that can be typed into the field. The options are:
 - **All** : Allows any type of character to be typed into the field
 - **Selected** : Defines a set of characters allowed. Using the configuration below, will be allowed to type letters, numbers and other characters (defined in the attribute More). The letters a, b and c (defined in the Less Attribute) are not allowed.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field.
- **Rows** : Allows to define the amount of records displayed in the field.
- **Width** : Defines the width the size of capture box (Capture Text).
- **Search options** : Allows to define the search settings of the field(Start equal to, Any part and End equals to).
- **Capture Text** : When not enabled, you will only view the field to inform the data. See an example below. lookup Settings configuration Interface.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

lookup Settings validation configuration Interface.

- **Show label with the description** : Displays a label with the description.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Control
- 4. Signature

Signature

General Settings

Configuration Interface of the Signature Field.

The signature field will help you creating more sophisticated forms and making it possible to store signatures in your database. Inside our development environment we have specific settings that will help you to customize your field, those options are:

- **Data Type** : You can define the type of field for the application. When it is defined as a text, it accepts letters, numbers and special characters.
- **Label** : Lets you define a label to the field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Background color** : Defines a color to the field background by using a valid hexadecimal color value or by choosing a color from the color palette.
- **Pen color** : Set a color to the pen by using a valid hexadecimal color value or by choosing a color from the color palette.
- **Width** : Lets you define a width to the field.
- **Height** : Set a height to the field.
- **Subtitle** : Defines the subtitle that will be displayed beside the field.
- **Initial Value** : Lets you define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
- **Save Variable** : Save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Set the name for the session variable that will receive the field value.
- **Disabled Field** : Define if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : Displays the HTML Object that will be used to display the field in the form.
- **SQL Type** : Displays the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of ScriptCase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **CSS of the Title**
 - **Font** : Select the font type, that will be applied to the application field title.
 - **Font Size** : Defines the font size, that will be applied to the application field title.
 - **Font Color** : Choose a color to the font by using a valid hexadecimal color value or from the color picker.
 - **Background Color** : You can define the color for the field by using a valid hexadecimal color value or from the color picker.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Set the underline style to the font.
 - **Border style** : Choose a style for the border.
 - **Collapse** : Defines the collapse for the border.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Choose a color for the border, using a color palette to apply to the title.
 - **Horizontal Alignment** : Position the label of the field in the wanted position (left,right,center and justify).
 - **Vertical Alignment** : Position the label of the field in the wanted position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : Defines a width for the title of the field.
 - **Height** : Set a height for the title of the field.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Control
- 4. Rating

Rating

General Settings

This field allows the developer to create a field for rating using stars (or any other image), where the final user will be able to select a rate option according to the configuration made by the developer.

Description of the attributes available for configuration.

Data type

Define the type of the field in the application. In this case we must select the type **Rating**.

Label

Define the application title that will be showed when the application is executed. The terminology used in the interface of your application is very important for the usability, and we need to use familiar terms for the final user, instead of use terms from the system.

Example, this field "rating" that is named **cmp_rating** however, the customer would have a better understanding if the label uses **Product Rating**.

Besides use a fixed text, the **Label** attribute allows the use of **Langs** to define the field title, allowing the [internationalization of your application](#).

Subtitle

Define the subtitle that will be shown in the field below the ratings. **Example**: "Thank you for your feedback!".

As in the **Label**, the **subtitle** attribute also allows the use of **Langs** for [internationalization of your application](#).

Amount of icons

Define the number of icons that will be displayed in the field. The value set in this attribute must be according to the way the evaluation will be realized.

To evaluate using Stars normally it is used five stars, so , we must to inform 5 in this attribute. After we set the images that will be displayed in the attribute **Enabled icon** and **Disabled icon** the result will be the same as in the example:

To use Like/Dislike **for example** we must inform 1 in the **amount of icons**, so only one image will be displayed and will change according to the click. In this case, the storage in the database will be 1 for like (enabled icon) and 0 for dislike (disabled icon). **You can see the field configured below as like/dislike:**

Enabled icon

Define the icon that will be shown for the amount selected. Following the like/dislike example, in this option, we must insert the Like image . We must do the same for a star evaluation, where for default we must insert the stars filled .

Disabled icon

Define the icon that will be shown for the not selected icons. In this attribute, we set an image to represent the opposite of the previous images set in the **Enabled icon**.

Still referring the example of the Like/Dislike, in this attribute we must set the Dislike .

As in the stars example, we must insert the image with the empty stars .

Initial Value (type)

Allow the definition of an initial value for the field when the form application is in insert mode. The only option available in this field is **Defined**. Selecting this option the attribute **Initial Value** will be showed for set the value.

Example: The initial value is 3 when inserting a new record, the rate field will initiate with 3 stars already selected.

The value defined in this attribute will overlap any value defined previously.

Save variable

Allow saving a session variable ([Global variable](#)) with the field value for being used in other applications.

Example: In the login form, the user name can be saved in the session and used to be displayed in the header of other applications.

Variable Name

In this attribute, we must define a name for the **session variable** that was enabled previously to receive the field value.

We must inform only the variable name, - **var_rating**.

We can recover the value as a [global variable](#).

HTML Type

HTML object used to show the form field.

SQL Type

Inform the type of the field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of ScriptCase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **CSS of the Title**

- **Font** : Select the font type, that will be applied to the application field title.
- **Font Size** : Defines the font size, that will be applied to the application field title.
- **Font Color** : Choose a color to the font by using a valid hexadecimal color value or from the color picker.
- **Background Color** : You can define the color for the field by using a valid hexadecimal color value or from the color picker.
- **Bold** : Applies the bold style to the font.
- **Underline** : Set the underline style to the font.
- **Border style** : Choose a style for the border.
- **Collapse** : Defines the collapse for the border.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Choose a color for the border, using a color palette to apply to the title.
- **Horizontal Alignment** : Position the label of the field in the wanted position (left,right,center and justify).
- **Vertical Alignment** : Position the label of the field in the wanted position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : Defines a width for the title of the field.
- **Height** : Set a height for the title of the field.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

Pages

A page is a container of blocks. Each application has at least one page by default. In Form, Control, and Search applications, there is possible to create many pages. The pages are like Tabs, where each tab contains one or more blocks.

Pages (available only in the Form, Control and Search applications) configuration Interface.

See the example below of the Form application using two Pages: General Data and Documents.

Form Application using Pages feature.

Pages Settings

The form application already comes with a default page, identified as "Pag1", but it is possible to rename it. Use the pages when you have an application that contains many fields. A form with more than 20 fields in a vertical way is challenging to use. So you could arrange the fields into the blocks and the blocks into the pages.

PAGES SETTINGS														
ATTRIBUTE	VALUE	DESCRIPTION												
<div style="border: 1px solid #ccc; padding: 5px; margin: 5px auto; width: 80%;"> <p style="text-align: center; margin: 0;">Edit the Page Settings</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Op</th> <th>Name</th> <th>Title</th> <th>Icon</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">🗑️</td> <td>Pag1</td> <td>Pag1</td> <td style="text-align: center;">🖼️</td> </tr> <tr> <td></td> <td>Pag2</td> <td>pag2</td> <td style="text-align: center;">🖼️</td> </tr> </tbody> </table> <p style="text-align: center; margin: 5px 0;">Include</p> </div>			Op	Name	Title	Icon	🗑️	Pag1	Pag1	🖼️		Pag2	pag2	🖼️
Op	Name	Title	Icon											
🗑️	Pag1	Pag1	🖼️											
	Pag2	pag2	🖼️											
Minimum tab width	<input type="text"/>													
Maximum tab width	<input type="text"/>													
Font	<input type="text"/> Aa													
Font Size	<input type="text"/>													
Selected Font Color	<input type="text"/>	🎨												
Selected Background Color	<input type="text"/>	🎨												
Non Selected Font Color	<input type="text"/>	🎨												
Non Selected Background Color	<input type="text"/>	🎨												

Pages Configuration Interface.

Including A New Page,

To include a new page, enter the information about the name and the label of the new page, select an image if you wish, then click on the button "Include".

Deleting A Page

To delete a page, click on the Trash icon corresponding to the page line.

Common Settings

Font

Set the font-family of the pages titles. By clicking on the right side icon, you can select the font-family from a list.

Font Size

Set the font size of the pages titles.

Selected Font Color

Set the font color of the selected page.

Selected Background Color

Set the Background Color of the selected page.

Non Selected Font Color

Set the font color of the non selected pages.

Non Selected Background Color

Set the Background Color of the non selected pages.

Blocks



Blocks are "containers" where you can position the application fields of Forms, Controls, or Slide Grids.

Scriptcase creates applications with one block by default. You can add more blocks as you wish, to organize it in the best way.

See below, the Columns Organization, and where you can define the position of the next block: beside or below the current one.

		Block	Title	Label			Fields		Organization		
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse	
Pag1											
✎ 🗑️	form_orders	form_orders	☐	☑	Beside ▾	1	Beside ▾	Below ▾	100%	Start open ▾	
Pag2											
✎ 🗑️	Block 2	Block 2	☐	☑	Beside ▾	1	Beside ▾	Below ▾	100%	Start open ▾	

Application Block configuration

On the left side of each block, there are two icons, the first one  to edit the information of the block and the second one  to delete the block.

Organizing The Position Of The Blocks

See below how to modify the display order of the Blocks in one Page.

Click and drag the block that you desire to modify to its new position.

Block		Title		Label		Fields			Organization	
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
		form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Pag2										
		Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open

Application Block Display configuration

- see how to remove a block from the display

Click on the block desired and drag it to the item "Blocks not Shown". This way, you can also drag the block to another page if desired. See the images below.

Block		Title		Label		Fields			Organization	
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
		form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Pag2										
		Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Blocks not shown										
		Block 3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open

Application Block Display configuration

Block		Title		Label		Fields			Organization	
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
		form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Pag2										
		Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Blocks not shown										
		Block 3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open

Application Block Display configuration

Block

- **ame:** The name that identifier the Block.
- **abel:** Title of the block to display in the application.

Title

- **isplay:** It controls the display of the block title.

Label

- **isplay:** It controls the display of the field labels of the block.
- **osition:** Options to display label :
 1. **bove:** Display the label above the field.
 2. **eside:** Display the label beside the field.
 3. **elow:** Display the label below the field.

Fields

- **olumns:** Amount of columns side by side in the block.
- **osition:** The way to display the fields in the block:
 1. **elow:** Display the fields one below the other respecting the number of columns.
 2. **eside:** Display the fields one beside the other respecting the number of columns.
 3. **ine:** Display the fields one beside the other with no tabulation.

Organization

- **ext:** The way to display the blocks in the page:
 1. **elow:** Set to show the following block below the current one.
 2. **eside:** Set to show the following block beside the current one.
 3. **abs:** Set to show the following block in a different tab then the current one.
- **idth:** Set the block width in pixels or percentages. Use the symbol "%" to indicates the value in percentage.
- **ollapse:** Enables the option to close the block.

Create A New Block

To include new blocks in an Application, click on the button . Then, enter the name and label of the block in the following interface and finish by click on Create.

Add New Block

NAME

LABEL

Creating application blocks configuration


Name

Name of the Block.

Label

Title of the block to display in the application.

Edit Blocks

To edit a block, click on the icon , that is on the left side of the block. Then you can see the following interface to define the parameters of the blocks. Click on Save to finish.

ATTRIBUTE	VALUE
Name	<input type="text" value="form_orders"/>
Title	<input type="text" value="form_orders"/>
Display Title	<input type="radio"/> Yes <input checked="" type="radio"/> No
Title Font	<input type="text" value="Aa"/>
Font Size	<input type="text" value=""/>
Font Color	<input type="text" value=""/>
Background Color	<input type="text" value=""/>
Background image	<input type="text" value=""/>
Title Height	<input type="text" value="20"/> pixels
Horizontal Alignment	<input type="text" value=""/>
Vertical Alignment	<input type="text" value=""/>
Display Label	<input checked="" type="radio"/> Yes <input type="radio"/> No
Columns	<input type="text" value="1"/>
Columns Width	<input type="text" value="Calculated"/>
Label Color	<input type="text" value=""/>
Fields Organization	<input type="text" value="Beside"/>
Label Position	<input type="text" value="Beside"/>
Next Block	<input type="text" value="Below"/>
Border Color	<input type="text" value=""/>
Border Width	<input type="text" value="0"/> pixels
Block Width	<input type="text" value="100%"/>
Block Height	<input type="text" value=""/>
Cell Spacing	<input type="text" value=""/> pixels
Collapse	<input type="text" value="Start open"/>

Application Block editing interface

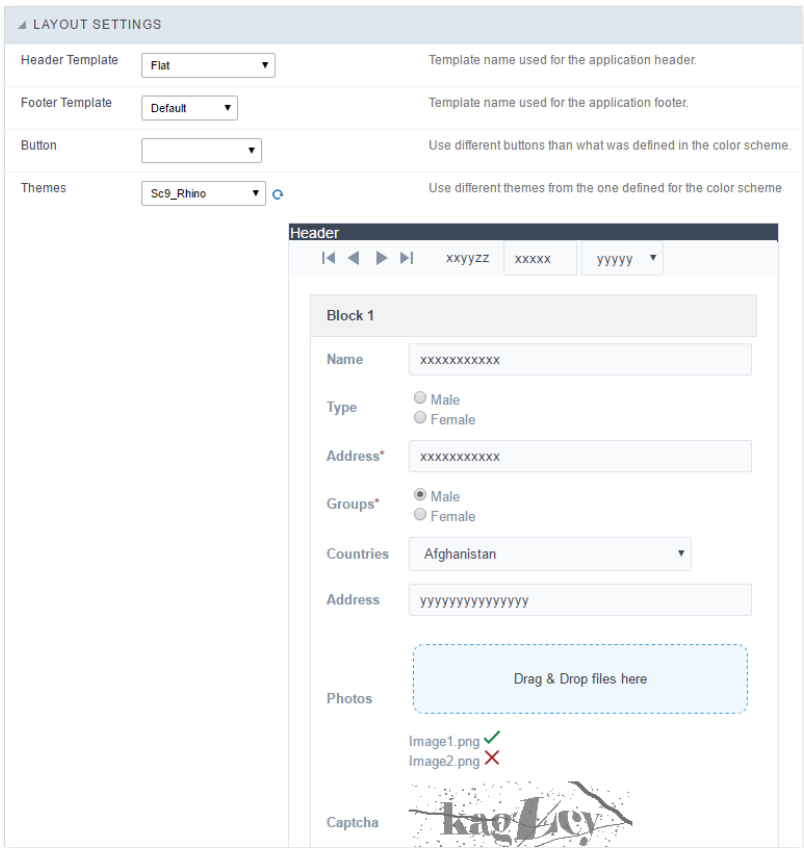
Name

Name of the block. ##### Title
 Block title for display. ##### Display Title
 This option, when active, allows displaying the block title. ##### Title Font
 Set the font family of the block title. ##### Font Size
 Set the font size of the block title. ##### Font Color
 Set the font color of the block title. ##### Background Color
 Set the Background Color of the block title. ##### Background image
 Set a Background image for the block title. ##### Title Height
 Height in pixels of the block title line. ##### Horizontal Alignment
 Horizontal Alignment of the block title (Left, Center, and Right). ##### Vertical Alignment
 Vertical Alignment of the block title (Top, Middle, and Bottom). ##### Display Label
 Display the labels of the fields in the block. ##### Columns
 Amount of field columns in a block. ##### Columns Width
 Set the field column width of the block. ##### Label Color
 Color of the field labels. ##### Fields Organization
 The way to display the fields in the block. ##### Label Position
 Set the position of the field labels of the block. ##### Next Block
 Set the position of the following block relating to the current one. ##### Border Color
 The border Color for the block. ##### Border Width
 The border Width for the block. ##### Block Width
 The width for the block. ##### Block Height
 The Height for the block. ##### Cell Spacing
 The Cell Spacing in the block. ##### Collapse
 It enables the option to close the block.

This module lets you editing display options for the application. You can apply visualization themes, organize blocks, define values, and the display format of the Header and Footer.

Layout Settings

ScriptCase creates an app by using default values defined per project, but you can change those values for a specific application on this interface.



Application Layout Settings

Header Template

Allows choosing the look of the application Header.

Footer Template

Allows choosing the look of the application Footer.

Button

Allows choosing the button theme for the application.

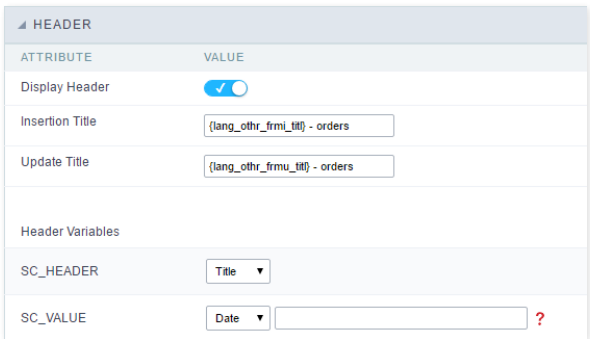
Themes

Choose one of the selected themes in the project properties. It defines the look of the application, like colors, fonts, and others.

Header & Footer

Header

Here we can define if we want to display the header or not, the application title, and fill the header variables. The header looks according to the selected template in the Layout settings interface.



Application Header Configuration

Display Header


This option determines if we want to display the header or not.

Title

Allows informing the title of the application.

Header Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **itle**: It displays the value of "**Application Title**" in the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Depending on the Application, you may have more than one title option.

Footer

FOOTER	
ATTRIBUTE	VALUE
Display Footer	<input type="checkbox"/>
Footer Variables	
NM_FOOTER_DATA	<input type="text"/>
NM_FOOTER_LOGO	<input type="text"/>
NM_FOOTER_TITLE	<input type="text"/>

Application Footer Settings


This page may change depending on the footer template chosen in the Layout Settings.

Display Footer

This option determines if we want to display the footer or not.

Footer Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Within the events you can include Scriptcase global and local variables, libraries and macros, PHP, JavaScript, HTML and CSS codes.

onApplicationInit

This event occurs only once, when the application is loaded and before the application executes the SQL. It can be used for data processing or checking variables.

Example: `if ([glo_var_dept] != 'financial'){ sc_redir(app_x.php); }`

OnScriptInit

This event occurs when the application scripts are started. It runs before the OnLoad events(when present), however, it will always be executed when running, reloading or navigating the application.

onLoad

This event runs when the application is finished loading.

onRefresh

This event runs when the refresh option of the application is enabled.

onValidate

This event runs when validating the information of the fields, when submitting the form.

onValidateFailure

This event runs when the form validation gives an error.

onValidateSuccess

This event runs when the form validation does not give an error.

OnClick

The ajax event OnClick is executed when the field that it's based on is clicked.

- reating a new ajax event



Creating a new ajax event

- electing a field

Creating Ajax Events	
Select the field to create an event	contacttitle
Select event for Ajax processing	onClick
Fields that will be passed as parameters Double click to check or uncheck	<ul style="list-style-type: none"> customerid companyname contactname contacttitle birthdate country regionid stateid city address postalcode phone
<input type="button" value="Create Event"/>	

Selecting a field of the ajax event

Choose a field to create an event To define in which field the event will be add to.

Choose an event that run the ajax Defines which event will be added to the field.

OnChange

The ajax event OnChange is executed when the value of the field that it's based on is modified.

- reating a new ajax event



Creating a new ajax event

- electing a field

Creating Ajax Events	
Select the field to create an event	contacttitle
Select event for Ajax processing	onClick
Fields that will be passed as parameters Double click to check or uncheck	<ul style="list-style-type: none"> customerid companyname contactname contacttitle birthdate country regionid stateid city address postalcode phone
<input type="button" value="Create Event"/>	

Selecting a field of the ajax event

Choose a field to create an event To define in which field the event will be add to.

Choose an event that run the ajax Defines which event will be added to the field.

OnBlur

The ajax event OnBlur is executed when the focus is removed from the field that event is based on.

- reating a new ajax event



Creating a new ajax event

- electing a field

Creating Ajax Events

Select the field to create an event

Select event for Ajax processing

Fields that will be passed as parameters
Double click to check or uncheck

- customerid
- companyname
- contactname
- contacttitle
- birthdate
- country
- regionid
- stateid
- city
- address
- postalcode
- phone

Create Event

Selecting a field of the ajax event

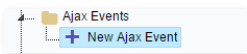
Choose a field to create an event To define in which field the event will be add to.

Choose an event that run the ajax Defines which event will be added to the field.

OnFocus

The ajax event OnFocus is executed when the field that it's based on is applied with a focus.

- reating a new ajax event



Creating a new ajax event

- electing a field

Creating Ajax Events

Select the field to create an event

Select event for Ajax processing

Fields that will be passed as parameters
Double click to check or uncheck

- customerid
- companyname
- contactname
- contacttitle
- birthdate
- country
- regionid
- stateid
- city
- address
- postalcode
- phone

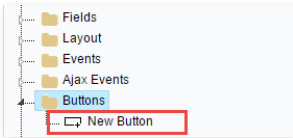
Create Event

Selecting a field of the ajax event

Choose a field to create an event To define in which field the event will be add to.

Choose an event that run the ajax Defines which event will be added to the field.

In addition to the applications buttons, you can also create new manual buttons. These buttons will be placed on the application toolbar.

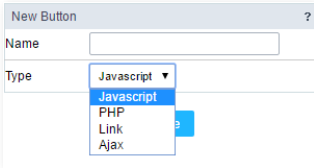


Creating new buttons

Creating a button

To create a new button, click the "new button" and type a name and the button type.

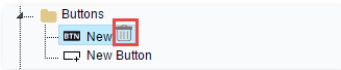
Form button types: (JavaScript, PHP, and Ajax).



Form button types

Deleting a button

To delete a button click on the icon next to the name of the button in the application menu (recycle bin).



Deleting a button

JavaScript

Display Mode

You can configure the display mode of the javascript button in Image, Button or Link.

Button

Button Settings: JavaScript	
ATTRIBUTE	VALUE
Display Mode	Button ▼
Label	JavaScript
Hint	
Confirmation Message	
Type	JavaScript

Setting up Javascript Button.

- Display Mode** You can select the display mode for the javascript button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Type** Description of the created button.

Image

Button Settings: JavaScript	
ATTRIBUTE	VALUE
Display Mode	Image ▼
Icon	<input type="text"/>
Hint	
Confirmation Message	
Type	JavaScript

Setting up Javascript Button.

- Display Mode** You can select the display mode for the javascript button in this option.
- Icon** Allows you to inform the icon that will be displayed on the button while the execution of the application.

- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Type** Description of the created button.

Link

► Button Settings: JavaScript

ATTRIBUTE	VALUE
Display Mode	Link ▼
Label	JavaScript
Hint	
Confirmation Message	
CSS Style	default ▼
Type	JavaScript

Setting up Javascript Button.

- Display Mode** You can select the display mode for the ajax button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- CSS Style** Name of the CSS class, style created in the layout editor.
- Type** Description of the created button.

Code Block

► JavaScript Code

Theme default ▼

```
1
```

Javascript button coding block.

In this block, only JavaScript is accepted.

PHP

Display Mode

You can configure the display mode of the PHP button in Image, Button or Link.

Button

► Button Settings: PHP

ATTRIBUTE	VALUE
Display Mode	Button ▼
Label	PHP
Hint	
Confirmation Message	
Reload quantity of records	No ▼
Type	PHP
Target	Same Window ▼

Setting up PHP Button.

- Display Mode** You can select the display mode for the link button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Reload quantity of records** This option is used to update the amount of records in the application.
- Type** Description of the created button.
- Target** Defines the window destination where the code will run (Open in the same Window, Open in another Window, Modal).

Image

ATTRIBUTE	VALUE
Display Mode	Image
Icon	<input type="text"/>
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
Reload quantity of records	No
Type	PHP
Target	Same Window

Setting up PHP Button.

- Display Mode** You can select the display mode for the link button in this option.
- Icon** Allows you to inform the icon that will be displayed on the button while the execution of the application.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Reload quantity of records** This option is used to update the amount of records in the application.
- Type** Description of the created button.
- Target** Defines the window destination where the code will run (Open in the same Window, Open in another Window, Modal).

Link

ATTRIBUTE	VALUE
Display Mode	Link
Label	PHP
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
CSS Style	default
Reload quantity of records	No
Type	PHP
Target	Same Window

Setting up PHP Button.

- Display Mode** You can select the display mode for the link button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- CSS Style** Name of the CSS class, style created in the layout editor.
- Reload quantity of records** This option is used to update the amount of records in the application.
- Type** Description of the created button.
- Target** Defines the window destination where the code will run (Open in the same Window, Open in another Window, Modal).

Code Block

ATTRIBUTE	VALUE
Code	*Ajax button coding block. *

In this block, you can use macros, PHP code and JavaScript.

Link

Display Mode

You can configure the display mode of the link button in Image, Button or Link.

Button

ATTRIBUTE	VALUE
Display Mode	Button
Label	Link
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
Type	Link

[Link](#)

Setting up Link Button.

Display Mode	You can select the display mode for the link button in this option.
Label	Text of the button that will be display in the application while executing.
Hint	Hint message for the button. (Displayed when the mouse hovers the button).
Confirmation Message	Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
Type	Description of the created button.

Image

ATTRIBUTE	VALUE
Display Mode	Image
Icon	<input type="text"/>
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
Type	Link

Setting up Link Button.

Display Mode	You can select the display mode for the link button in this option.
Icon	Allows you to inform the icon that will be displayed on the button while the execution of the application.
Hint	Hint message for the button. (Displayed when the mouse hovers the button).
Confirmation Message	Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
Type	Description of the created button.

Link

ATTRIBUTE	VALUE
Display Mode	Link
Label	Link
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
CSS Style	default
Type	Link

Setting up link Button.

Display Mode	You can select the display mode for the ajax button in this option.
Label	Text of the button that will be display in the application while executing.
Hint	Hint message for the button. (Displayed when the mouse hovers the button).
Confirmation Message	Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
CSS Style	Name of the CSS class, style created in the layout editor.
Type	Description of the created button.

Setting Up The Link

- electing the applications

Link between applications - Application List

Select the application that will be called:

Application: All By folder By type

- calendar_events
- chart_customers
- dashboard
- form_customers
- form_employees
- form_orders
- form_sec_users
- grid_categories

Choosing the application for the button link.

You should select an application to be called from the button link.

- ink Parameters

Choosing the parameters for the button link.

Field Allows you to use an existing field from the current application as a parameter for the link.

Variable Allows you to use a global variable from the current application as a parameter for the link.

Fixed Allows you to inform a fixed value as a parameter for the link.

Empty No value will be passed as a parameter for the link.

- ink Properties (Grid)

Configuring the properties for the link button when the destined application is a Grid.

Link Operation Mode How the link will open (Open in the same Window, Open in another Window, Modal).

Exit URL for the target application URL or an application that be redirect to when exiting the Grid application.

Initial Mode Allows you to define the initial mode of the grid application (Search or Grid).

Number of Lines Allows you to define the amount of lines displayed in the Grid.

Number of Columns Allows you to define the amount of columns displayed in the Grid.

Paging Enable the paging in the Grid.

Display Header Enable the Grid Header.

Active Navigation Buttons Enable the navigation button (First, Back, Next and Last) in the Grid.

- ink Properties (Form)

Configuring the properties for the link button when the destined application is a Form.

Link Operation Mode How the link will open (Open in the same Window, Open in another Window, Modal).

Exit URL for the target application URL or an application that be redirect to when exiting the Form application.

Enable insert button on target application Enable the "New" button in the Form Application.

Enable update button on target application Enable the "Update" button in the Form Application.

Enable delete button on target application Enable the "Delete" button in the Form Application.

Enable navigation button on target application Enable the navigation button (First, Back, Next and Last) in the Form.

Enable button to edit a grid record Enable the buttons that allow you to edit the records of a Grid

Ajax

Display Mode

You can configure the display mode of the Ajax button in Image, Button or Link.

Button

► Button Settings: Ajax


ATTRIBUTE	VALUE
Display Mode	Button ▼
Label	Ajax
Hint	
Confirmation Message	
Type	Ajax
Show in Update mode	<input checked="" type="radio"/> Yes <input type="radio"/> No
Show in the inserting mode	<input type="radio"/> Yes <input checked="" type="radio"/> No

Setting up Ajax Button.

- Display Mode** You can select the display mode for the Ajax button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Type** Description of the created button.

Image

► Button Settings: Ajax

ATTRIBUTE	VALUE
Display Mode	Image ▼
Icon	<input type="text"/> 
Hint	
Confirmation Message	
Type	Ajax
Show in Update mode	<input checked="" type="radio"/> Yes <input type="radio"/> No
Show in the inserting mode	<input type="radio"/> Yes <input checked="" type="radio"/> No

Setting up Ajax Button.

- Display Mode** You can select the display mode for the Ajax button in this option.
- Icon** Allows you to inform the icon that will be displayed on the button while the execution of the application.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Type** Description of the created button.

Link

► Button Settings: Ajax

ATTRIBUTE	VALUE
Display Mode	Link ▼
Label	Ajax
Hint	
Confirmation Message	
CSS Style	default ▼
Type	Ajax
Show in Update mode	<input checked="" type="radio"/> Yes <input type="radio"/> No
Show in the inserting mode	<input type="radio"/> Yes <input checked="" type="radio"/> No

Setting up Ajax Button.

- Display Mode** You can select the display mode for the ajax button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- CSS Style** Name of the CSS class, style created in the layout editor.
- Type** Description of the created button.

Code Block

► PHP Code ?

Theme default ▼

1

Ajax button coding block.

In this block, you can use macros, Ajax code and JavaScript.

Settings

With this interface, you can set the common attributes of the app.

ATTRIBUTE	VALUE
Application Code	grid_customers (9.00.0000)
Description	<input type="text"/>
Documents Path	C:/Program Files/NetMake/v9/www/
Image Directory	/scriptcase/file/img
Application images	<input type="button" value="Add"/> <input type="button" value="Remove"/>
Language	English (United States) ▼
Share Location Variable	<input checked="" type="checkbox"/>
Charset	<input type="text"/>
Share Theme Variable	<input checked="" type="checkbox"/>
Folder	root ▼
Edit by Project	<input checked="" type="checkbox"/>
Timeout	<input type="text" value="0"/>
HelpCase Link	Application ▼ Search ▼ Summary ▼

Application Settings Interface

- Attributes**
 - Application Code** : It is the name that defines an application. An app can be renamed at the [List of Application](#).
 - Description** : This field contains a brief description of the application objectives.
 - Documents Path** : The absolute path to store uploaded documents in the application.
 - Image Directory** : The filesystem directory to store the application images.
 - Application images** : Import images into the application to allow using them in the application.
 - Language** : Set the default language of the application. Display all the application hints and messages in the selected language.
 - Share Location Variable** : Define if the app shares the regional settings with other applications through a session variable.
 - Charset** : Define a specific charset to use in the application.
 - Share Theme Variable** : Define if the app shares the Theme settings with other applications through a session variable.
 - Folder** : Define the project folder that contains the app.
 - Edit by Project** : Define if other project developers can edit the application.
 - Timeout** : Set the session runtime timeout in seconds. If the value is Zero, it assumes the default timeout of the PHP.
 - HelpCase Link** : It allows to associate a [HelpCase](#) file with the application.

Notification Settings

ATTRIBUTE	VALUE
Use SweetAlert	<input type="checkbox"/>
Error Position on the field	Down ▼
Show the Error Title in the Application	<input checked="" type="checkbox"/>
Show the Error Title in the Field	<input type="checkbox"/>
Error Title	{lang_errm_errt}
Script Error	<input type="checkbox"/>
SQL Error	<input checked="" type="checkbox"/>
Debug Mode	<input type="checkbox"/>
Ajax Error Output	<input checked="" type="checkbox"/>

- Use SweetAlert**: Use the SweetAlert to display messages from the application. When this option is active, it will replace the browser's "confirm" and "alert".
- Use SweetAlert position using Toast** : The position to display error messages on the application.

- **error Position on the field** : The position to display error messages when criticizing the field.
- **how the Error Title in the Application** : Define to display the title line of the error message or not.
- **how the Error Title in the Field** : Define to display the title line of the error message in the field or not.
- **cript Error** : Allows displaying the line code where there is an error..
- **QL Error** : Allows displaying the SQL statement if it got an error.
- **ebug Mode** : Runs the application in Debug mode, showing all SQL statements the application is executing.
- **jax Error Output** : Enables the Ajax alert for debugging errors.

Navigation

This interface allows defining the navigating behavior of the application

NAVIGATION	
ATTRIBUTE	VALUE
Exit URL	<input type="text"/> 🔗
Close on Exit	<input type="checkbox"/>
Redirect URL	<input type="text"/> 🔗
Redirect Variable	<input type="text"/>

Navigation Interface.

Exit URL

URL to where the user goes when he clicks on the “exit” button.

Close on Exit

Close the browser window when the user clicks on the “exit” button.

Redirect URL

Redirect to another URL in case there aren't any global variables available.

Redirect Variable

Creates a variable with the application name and sends it to the redirected application.

Messages

On this screen, you can redefine the default application messages to the end-user by customizing the validation messages for Insert, Update, and Delete. You can also define messages for SQL errors and the confirmation outputs.

Messages

Messages		
ATTRIBUTE	VALUE	DESCRIPTION
No Records Message	<input type="text"/>	When the application has no records.it will display this customized text.
Primary key violation	<input type="text"/>	Message to display when the primary key constraint is violated
Unique key violation	<input type="text"/>	Message to display when the unique constraint is violated

No Records Message

The message displayed when the application has no records.

Primary key violation

The message displayed when there is a violation in the database's primary key constraint.

Unique key violation

The message displayed when there is a violation in the database's unique constraint.

Insertion Messages

Insertion messages		
ATTRIBUTE	VALUE	DESCRIPTION
Message After Insert	<input type="text"/>	Message to display after inserting a record.
Message to Confirm Insert	<input type="text"/>	Message to display to confirm inserting of a record.

Message After Insert

The message displayed when inserting a new record.

Message to Confirm Insert

The message displayed if the end-user wants to confirm the inserting of a new record.

Update Messages

Update messages		
ATTRIBUTE	VALUE	DESCRIPTION
Message After Update	<input type="text"/>	Message to display after updating a record.
Message to Confirm Update	<input type="text"/>	Message to display on the update confirmation

Message After Update

The message displayed when updating a record.

Message to Confirm Update

The message displayed if the end-user wants to confirm the changes of a record.

Messages Of Delete

Messages of Delete.		
ATTRIBUTE	VALUE	DESCRIPTION
Message After Delete	<input type="text"/>	Message to display after delete a record.
Message to confirm Delete	<input type="text"/>	Message to display on the delete confirmation

Message After Delete

The message displayed when deleting a record.

Message to confirm Delete

Displays a customized message asking to confirm the record deletion.

If you make no changes, the application will use the default values for the messages. Those values can be configured in [Locales -> Application Language](#).

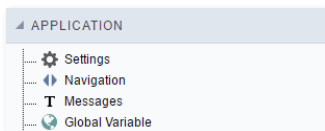
Global variables

This screen shows all global variables used in the application

The global variable is an external parameter required for running the application. The application can include global variables in the WHERE clause, field definitions and names, event programming, and so on.

You can define global variables by using square brackets ([variable]). You can pass the parameters to the application through one of the methods: Session, Post, and Get

if you need to use [Database Identifiers](#) in or tables, we recommend using the double quotation marks(" ") instead of the brackets([]) because of the conflict with the [Global Variables syntax](#) of Scriptcase. Using the brackets as database identifiers may lead to problems in the Scriptcase applications.



Global variables Interface.

Example:

```
Select CustomerID, CustomerName from Customers WHERE CustomerID = '[v_customerid]'
```

We do not recommend using session variables (global variables) with the same name as the table fields.

In this case, the variable **v_customerid** is displaying in the global variables configuration.

VARIABLE SETTINGS	
ATTRIBUTE	VALUE
global	<div style="border: 1px solid black; padding: 5px;"> <p>Scope</p> <p><input type="checkbox"/> SESSION</p> <p><input checked="" type="checkbox"/> POST</p> <p><input checked="" type="checkbox"/> GET</p> </div> <div style="border: 1px solid black; padding: 5px; margin-top: 5px;"> <p>Settings</p> <p><input type="checkbox"/> Optional</p> </div> <div style="border: 1px solid black; padding: 5px; margin-top: 5px;"> <p>Type</p> <p><input type="radio"/> Out</p> <p><input checked="" type="radio"/> In</p> </div>

Global variables configuration Interface.

Attribute

It shows the variable's name in the application.

Value

Defines the behavior of the variables, divided into three blocks, they are:

Scope

Defines how the application receives the variable. A variable defined as the POST method in the app does not receive value if it comes with the GET method. Those are the methods available:

- **ession**: Defines that the variable must be created as a PHP session variable by another application.
- **ET**: Defines that the variable must be passed by the URL, that is, visible by the browser.
- **OST**: Defines that the variable must be passed through the POST method of an HTML form.

Settings

- **ptional**: Check it to don't validate the variable at runtime.

Type (In / Out)

Defines whether the variable comes from another application (In), or it is starting in the current app (Out).

Description

Indicates where the application is using the variable.



1. [Home](#)
2. Applications

General Overview

Allows to create link between applications that are in the same project, the link option incorporates various functionalities like above.

New Link

To create a new link you just need to access the option **Link between Applications** located in the menu at the left side.

When clicking in **New Link** the screen below will be display:

When clicking in **New Link** the screen to choose the type of link that will be created will be displayed.

[Edit Link:](#)

Allows to edit registers of the grid using a form.

[Capture Link:](#)

Allows to create a link specifically to the filter's fields, capturing data from other grid.

[Field Link:](#)

Allows to create a link represented through a link, from a grid's column to any application from the project.

[Button Link:](#)

Allows to create a link through grids buttons to any application from the project.

1. [Home](#)
2. [Applications](#)
3. Control
4. Application link

Application link

Creating an Application Link

This type of link allows the developer to create a link from a grid to a form with the objective of editing the register of a grid's row.

In the link options, we will choose the **Application Link**. When we choose this option, edit a register from a Grid will be possible.

Applications list

After we select this option, will be displayed the applications list to what you want to create the link.

This screen can be viewed from the following ways:

All:

In which is possible to see all the project's applications. **Example:**

By folder:

In which is possible to see the applications according to the folders in which each of them are.

By type:

In which is possible to see the applications grouped by its respective types.

Definition of Parameters

In this setting we are going to define the type of parameter that will be passed to the next application.

We have two options at **Type of Parameters**, they are:

Fixed value:

This option allows the user to define a fixed value that will be used to the call of the next application.

No value:

This option allows to create the link without the need to send any parameter to the next application.

It is also possible to see a refresh button at its right side.

This button should be used when a new parameter is added to the target application, so the new parameter can be loaded in the current application to make the link.

Link properties

In this screen we will set the application display mode that will be called in the link.

There are five display options, they are:

Open in the same window:

When we use this option the target application will be displayed in the same window of the application, and the target application will have a back button so we can return to the previous application.

Open in another window:

When we use this option the target application will be displayed in other browser window, and the target application will have a exit button so we can close this window.

Open in an iframe:

When we use this option the target application will be displayed in the same window of our application, however it will be displayed in a iframe in one of the four options available in the iframe settings.

Open in a parent:

When we use this option the target application will be displayed in the same window of our application, and the target application will have a back button so we can return to the previous application.

Modal:

When we use this option the target application will be displayed in the same window of our application, however it will be displayed in a small window above the current application.

Open in the same window.

When we use this option the target application will be displayed in the same window of the application, and the target application will have a back button so we can return to the

previous application.

When we select this option, we can set these following options:

Open in another window

When we use this option the target application will be displayed in another browser window.

When we select this option, we can set the following options:

Modal

When we use this option the target application will be displayed in the same window of the application, however she will be displayed in a small window above the current application.

When we select this option, we can set the following options:

Height:

Allows to set the Modal height.

Width:

Allows to set the Modal width.

1. [Home](#)
2. [Applications](#)
3. Control
4. Button link

Button link

Creating a Button Link

Allows the developer to create a link where the call to the other application will be done through a button.

In the type of links options, we will choose the **Button Link**. Choosing this option it will be possible to create a link to any other application.

List of applications

After selecting this option, The list of applications to what you want to create a link will be displayed.

This screen can be viewed in the following ways:

All:

In which is possible to see all the project's applications. **Example:**

By folder:

In which is possible to see the applications according to the folders in which each of them are.

By type:

In which is possible to see the applications grouped by its respective types.

Definition of Parameters

In this setting we are going to define the type of parameter that will be passed to the next application.

We have two options at **Type of Parameters**, they are:

Fixed value:

This option allows the user to define a fixed value that will be used to the call of the next application.

No value:

This option allows to create the link without the need to send any parameter to the next application.

It is also possible to see a refresh button at its right side.

This button should be used when a new parameter is added to the target application, so the new parameter can be loaded in the current application to make the link.

Link properties

In this screen we will set the application display mode that will be called in the link.

There are five display options, they are:

Open in the same window:

When we use this option the target application will be displayed in the same window of the application, and the target application will have a back button so we can return to the previous application.

Open in another window:

When we use this option the target application will be displayed in other browser window, and the target application will have a exit button so we can close this window.

Modal:

When we use this option the target application will be displayed in the same window of our application, however it will be displayed in a small window above the current application.

Open in the same window

When we use this option the target application will be displayed in the same window of the application, and the target application will have a back button so we can return to the previous application.

When we select this option, we can set the following options:

Exit URL for the target application:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Hint for the link:

In this option we will inform a message that will be displayed when the mouse cursor is over the button with the link.

Open in another window

When we use this option the target application will be displayed in another browser window.

When we select this option, we can set the following options:

Exit URL for the target application:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Close the form after an update:

Choosing the option **Yes**, the window will be closed after doing the update of a register.

Close the form after an insert:

Choosing the option **Yes**, the window will be closed after doing the insert of a register.

Hint for the link:

In this option we will inform a message that will be displayed when the mouse cursor is over the button with the link.

Open the application in a tab inside the menu:

Choosing the option **Yes**, the application will be open in **amenu tab**, when executed from a menu application, instead of opened in another browser window.

Choosing the option **Yes**, there are also some other settings to be displayed:

Title to the tab:

This option allows the developer to set a title to the tab that will be opened when used in a Menu application.

Hint to the tab:

This option allows the developer to set a message to be displayed when the mouse cursor is over the tab Menu.

Active tab icon:

This option allows the developer to set an icon to be displayed in the tab when used in a menu application.

Modal

When we use this option the target application will be displayed in the same window of the application, however she will be displayed in a small window above the current application.

When we select this option, we can set the following options:

Height:

Allows to set the Modal's height.

Width:

Allows to set the Modal's width.

Exit URL for the target application:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Close the form after an update:

Choosing the option **Yes**, the window will be closed after doing the update of a register.

Close the form after an insert:

Choosing the option **Yes**, the window will be closed after doing the insert of a register.

Hint for the link:

In this option we will inform a message that will be displayed when the mouse cursor is over the button with the link.



1. [Home](#)
2. [Applications](#)
3. Control
4. Link edit

Link edit

In the Link Folder of the Application Menu (Image Below) are displayed the links existing in the application and also the item New Link. When clicking on the existing link it is displayed the screen below that allows to manage the links.

 *Editing Links.*

Actions

Properties Change the link's behavior, position, and how the link opens.

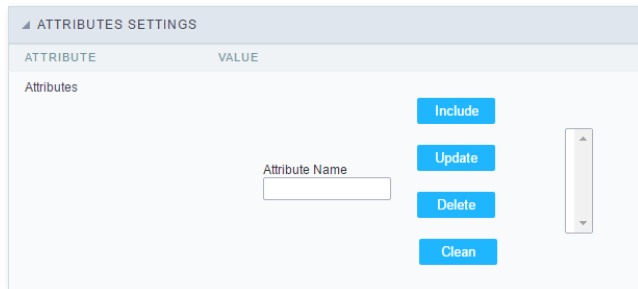
Link Change the application that's being called in the link their parameters.

Delete Remove the existing link.

ScriptCase has incorporated the concept of Object Oriented programming, using attributes, resources, methods and libraries. It is possible to create your own business rules in applications, and by using these concepts you can reap huge rewards in terms of better organization and improved development.

Attributes

The attributes are variables that has the global scope declared in the application. An attribute that can be referenced in all the methods and the application's events.

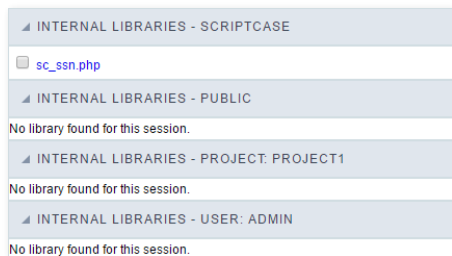


Attributes configuration Interface.

- **attributes**
 - **attribute Name** : This field allows to inform the attributes name, after typing the name just click on **include**. In case you want to **update** click on the attributes name (located on the buttons' right side), do the necessary modifications and click on update. In case it is necessary to **delete** an attribute, it is necessary to do same process as the update, but when selecting the attribute, it is necessary to click on the delete button. The **clear** button, as the name suggests, it clears the list of all the attributes listed on the right.

Internal libraries

To use the internal libraries just select the correspondent (Project Libraries for example). This process allows the methods existing in the library, to be visible in all the application's Events and Methods. See how to manage the libraries by [clicking here](#).

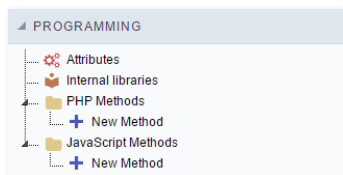


Internal Libraries management Interface

PHP Methods

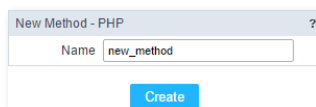
Methods are function or procedures declared by the developer, that helps when applying the routines. Using methods in the applications allows to reuse your code throughout the application, optimizing the development experience.

Creating A New Method



- Give the method a name and click Create. As in the image below.

PHP methods are incorporated into the same class, so it is not possible to use two functions with the same name even when used in different methods.



- Methods can receive parameters.

```
function new_method
1 echo "Hello World!!!";
```

- dd the amount of variables:

Definition of the parameters of the method:

new_method

- No defined parameter.

Add 1 Parameter(s) Cancel

Save

- refining the variables:

Insertion of Parameters

Name	Type	Value Standard
	For Value	
	For Value	
	For References	

Save Back Cancel

- **ame** : Type in the variable's name.
- **ype** : Selecting the type of variables: For Value or For Reference.
- **alue Standard** : The parameter's value used to initialize when calling the method.

References allows to create a second name for a variable that you can use to read and modify the variable's original information.

- diting a parameter:

Definition of the parameters of the method:

new_method

Parameters
Stest = test

☑ ☐ ✎ ✕

Add 1 Parameter(s) Cancel

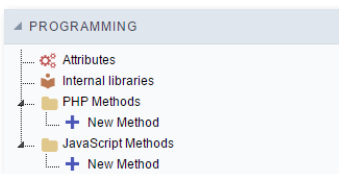
Save

- ☑ : Checking all the parameters in the parameter list.
- ☐ : Uncheck all the parameters in the parameter list.
- ✎ : Edit the selected parameter of the list.
- ✕ : Deletes the selected variable of the list.

JavaScript Methods

Methods are function or procedures declared by the developer, that helps when applying the routines. Using methods in the applications allows to reuse of the your code throughout the application, optimizing the development experience.

Creating A New Method



JavaScript method creation Interface

- define a name for the method and click on Create. Like the image below.

New Method - JAVASCRIPT ?

Name

Include Method.

- methods can receive parameters.

function new_method

```
1 echo "Hello World!!!";
```

Definition of the parameters of the method:

new_method

No defined parameter.

Parameter(s)

- add the amount of variables:

JavaScript method parameter definition

- defining the variables:

Insertion of Parameters

Name	Type	Value Standard
<input type="text"/>	For Value	<input type="text"/>

Defining the variables

- **name** : Type in the variable's name.
- **type** : Selecting the type of variables: For Value or For Reference.
- **value Standard** : The parameter's value used to initialize when calling the method.

References allows to create a second name for a variable that you can use to read and modify the original variable's information.

- editing a parameter:

Definition of the parameters of the method:

new_method

Parameters

Stest = test

Parameter(s)

- : Checking all the parameters in the parameter list.
- : Uncheck all the parameters in the parameter list.
- : Edit the selected parameter of the list.
- : Deletes the selected variable of the list.

This option is available in the Control application if the item **Set the template HTML manually** template is selected when creating the application.

APPLICATION DATA		THEME
Connection *	<input type="text" value="conn_example"/>	Name * <input type="text" value="control_2"/>
Localization *	<input type="text" value="English (United States) - UTF-8"/> <input type="button" value="📄"/>	
<input checked="" type="checkbox"/> Set the template HTML manually		

To use the custom **HTML** in Control, depends on specific **Markups** so that the Scriptcase can include its control and validation routines.

SETTINGS

ATTRIBUTE	VALUE	DESCRIPTION
<div style="display: flex; justify-content: space-between; border-bottom: 1px solid #ccc; padding-bottom: 5px;"> Check HTML CSS Javascript External library Preview Create fields </div> <p>The use of custom HTML in the Control Application depends on specific markups, so that Scriptcase can include its control and validation routines.</p> <p>Click in the "Check" button to verify if the HTML has all markups needed.</p>	<pre> 1 <html> 2 3 <head> 4 <!--SC_PAGE_CHARSET--> 5 <!--SC_JS_LIB--> 6 <link rel="stylesheet" type="text/css" href="control_1.css" /> 7 <script type="text/javascript" src="control_1.js"></script> 8 <title><!--SC_PAGE_TITLE--></title> 9 </head> 10 11 <body> 12 <Form {SC_FORM_ATTR}> 13 <!--SC_FORM_HIDDEN--> 14 <!--SC_FIELD_LABEL_my_field--> 15
 16 <input {SC_FIELD_INFO_my_field} class="{SC_FIELD_CLASS}" type="text" /> 17
 18 <input type="button" {SC_FORM_SUBMIT_INFO} /> 19 </Form> 20 </body> 21 22 </html> </pre>	

Check

Check
HTML
CSS
Javascript
External library
Preview

TIPS

- scDisplayUserError()
- SC_LIB_SCRIPTCASE
SC_LIB_PUBLIC
SC_LIB_PROJECT
- SC_IMG_URL
- CSS file
- JavaScript file

REQUIRED

- SC_FIELD_CLASS ✓
- SC_FORM_ATTR ✓
- SC_FORM_HIDDEN ✓
- SC_FORM_SUBMIT_CALL
SC_FORM_SUBMIT_INFO ✓
- SC_JS_LIB ✓
- SC_PAGE_CHARSET ✓

OPTIONAL

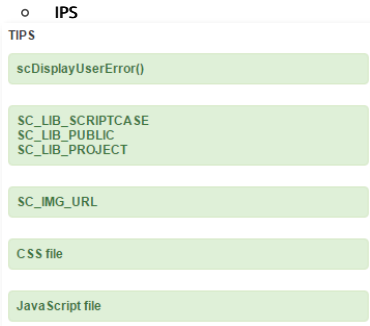
- SC_PAGE_TITLE ✓

Clicking the **Check** button, it verifies the structure to confirm which **Markups** were used in the content of the **HTML**.

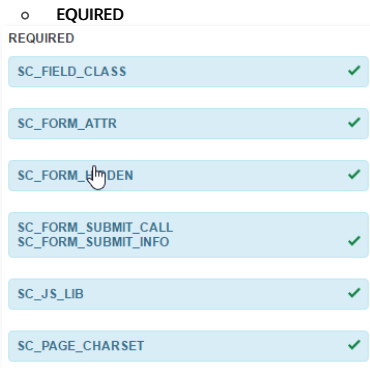


When clicking on the **Markups** of the **Check** item, a modal window is displayed with tips for using the same **Markup**.

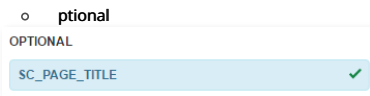
- The **Markups** are divided in 3 different categories:



These **Markups** are not required, however, they are used to import files from libraries, images and other features.

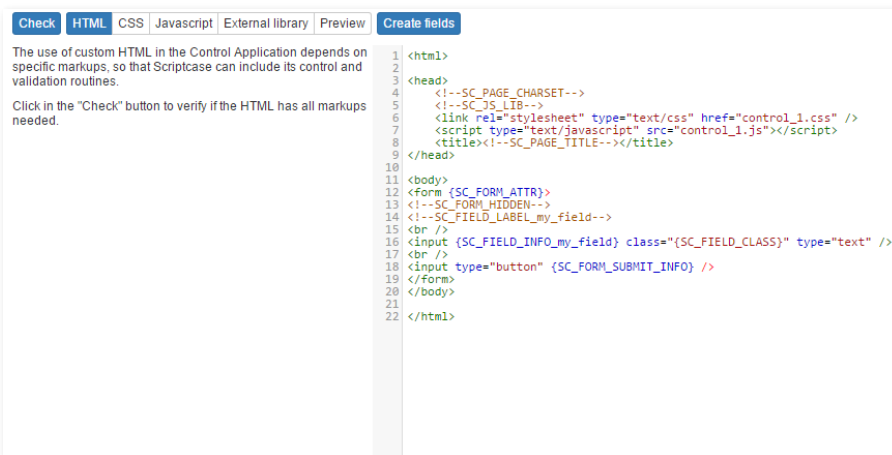


These **Markups** are of obligatory use in the item **HTML**. If they are not used, certain internal Scriptcase routines will not be executed.



Like the category name, these **Markups** are optional, not influencing the correct execution of the application.

HTML



In this item, we can insert the desired HTML, remembering to use the required **Markups**.

The `control.js` and `control.css` files refer to the contents of the `JavaScript` and `CSS` items respectively. By default, they are automatically created by the application, by default: `appName.js` and `appName.css`.

CSS

The screenshot shows the Scriptcase interface with the 'CSS' tab selected. The left pane contains the following text:

Check HTML CSS Javascript External library Preview Create fields

The use of custom HTML in the Control Application depends on specific markups, so that Scriptcase can include its control and validation routines.

Click in the "Check" button to verify if the HTML has all markups needed.

The right pane shows a vertical scrollbar with the number '1' at the top.

In this item, we can use our custom CSS to take effect in the `HTML` code used in the previous item.

JavaScript

The screenshot shows the Scriptcase interface with the 'Javascript' tab selected. The left pane contains the same text as the CSS tab. The right pane shows a code editor with the following JavaScript code:


```
1 function scDisplayUserError(errorMessage) {
2   alert(errorMessage.replace("<br />", "\n"));
3 }
```

In this item, we can use our custom `JavaScript` to take effect in the code `HTML` used in the previous item.

External Libraries

In this option, we can define if we will use the files `.HTML`, `.CSS` and `.JS` from an external library or if we will use the code `HTML`, `CSS` and `JavaScript` of the items of the application itself.

- o not use external library

The screenshot shows the Scriptcase interface with the 'External library' tab selected. The left pane contains the same text as the previous tabs. The right pane contains the following text:

You can choose to use an external library from Scriptcase instead of saving the HTML content in the application itself. By doing this, you have the possibility to use multiple CSS and Javascript files, in addition to images.

To create/edit an external libraries you can access the main menu from Scriptcase using the option: Tools > External libraries. Once created and associated with the current project, you can choose the library to use in this application.

Below the text are two radio buttons:

- Do not use external library
- Use external library

This option defines that the `HTML`, `CSS`, and `JavaScript` code to be used,

come from the current application's `HTML`, `CSS`, and `JavaScript` items.

- se external library

Check
HTML
CSS
Javascript
External library
Preview
Create fields

The use of custom HTML in the Control Application depends on specific markups, so that Scriptcase can include its control and validation routines.

Click in the "Check" button to verify if the HTML has all markups needed.

You can choose to use an external library from Scriptcase instead of saving the HTML content in the application itself. By doing this, you have the possibility to use multiple CSS and Javascript files, in addition to images.

To create/edit external libraries you can access the main menu from Scriptcase using the option: Tools > External libraries. Once created and associated with the current project, you can choose the library to use in this application.

Do not use external library

Use external library

No HTML file was found in the root directory of the external libraries that are associated with this project. There needs to be an HTML file to use the external library in the Control Form. The file extension can be either htm or html.

Choose the HTML file from an external library that will be used as the body of the Control application

Project
info (no HTML files)

When selecting this option, we define the use of external libraries that contain the **HTML**, **CSS** and **JavaScript** to be used by the application. The **HTML**, **CSS** and **JavaScript** items will be disabled.

Preview

Cancel

You will see how the application will appear after its generation.

Create Fields

SETTINGS
ATTRIBUTE VALUE

Check
HTML
CSS
Javascript
External library
Preview
Create fields

The use of custom HTML in the Control Application depends on specific markups, so that Scriptcase can include its control and validation routines.

Click in the "Check" button to verify if the HTML has all markups needed.

```

1 <html>
2
3 <head>
4 <
5 <
6 <
7 <
8 <
9 </head>
10
11 <body>
12 <form>
13 <!--S
14 <!--S
15 <br />
16 <input
17 <br />
18 <input type="button" (SC_FORH_SUBMIT_INFO) />
19 </form>
20 </body>
21
22 </html>

```

Create fields

Please enter the type and label of the following fields that were found within the HTML:

Field	Data Type	Label
my_field	Text	My Field

Cancel
Save

If you use field Markup before the field is created in the menu item **Fields > New Field**, and the item **(Create Fields)** is pressed, a modal window is displayed to synchronize the fields, after confirmation, the fields will be displayed in the **Fields** menu item.

When adding fields from the menu item **Campos > New Field**, Scriptcase will automatically insert new mandatory **Markups** related to the newly created field to be used in the **HTML** item.

CONTROL

- User HTML
- Settings
- Edit Fields
- SQL
- Security
- Log
- Fields
 - + New Field
- Events
- Ajax Events

SETTINGS

ATTRIBUTE	VALUE
Check	HTML
CSS	
Javascript	
External library	
Preview	

TIPS

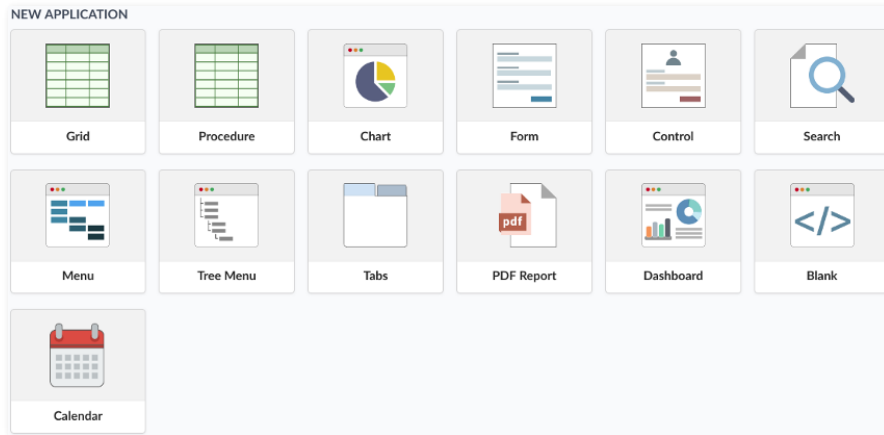
- scDisplayUserError()

SC_LIB_SCRIPTCASE
SC_LIB_PUBLIC
SC_LIB_PROJECT

New Application

The creation process of one application is simple enough and intuitive.

You just need to select which application you want to generate in the applications available in Scriptcase, That it will be generated automatically.



Application Data

When the application is selected, the following screen will be displayed below.

In this table you need to select the connection that will be used in the application. The tables will be loaded and displayed in the field **Table** when the connection is selected.

It is possible to select more than one table in the application creation.

- **connection** - It defines the used connection to create the application. All the project's application will be displayed.
- **ables** - It defines the used tables in the application. (Form and Calendar can only use one table).
- **ields** - It defines the fields that will be part of the applications.
- **QL Select Statement** - Display the select statement created after selecting the tables and fields. This field allows also to insert a previously created SQL, if the used tables are available in the database selected in the connection.

Theme

In this screen you select the theme that the application will use. The default project's theme, that can be checked and changed in **Project > Properties**, will be selected automatically.

APPLICATION DATA RELATIONSHIP EDIT FIELDS **THEME**

Sc9_Rhino ▾

Header

◀ ◀ ▶ ▶ Add Save

Block 1.1

Title 1 Object text

Block 2.1

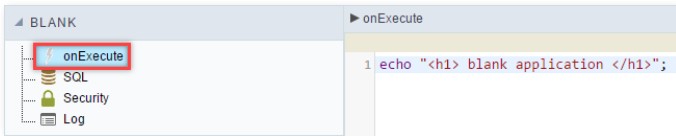
Title 1	Title 2	Title 3
11111111	22222222	33333333
11111111	22222222	33333333

Footer

- **heme** - List of available themes in the project.

*After all the settings are done, click in the button **Create**. In the next screen the settings of the created application will be displayed.*

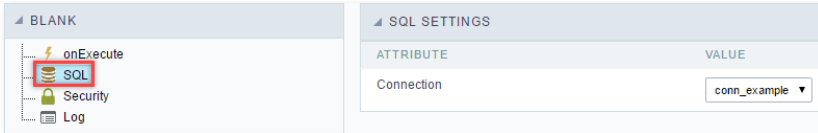
In this menu is entered the PHP code that will be executed by the application.



OnExecute.

- **nExecute** : This event allows to entry PHP commands, JavaScript, CSS, HTML, ScriptCase macros and libraries.

In blank applications you can use one connection to access the database by default but as in other application types (Forms, Grids and Control) you can run SQL commands using other connections different from the native application connection.



SQL Settings.

- **onnection** : This option allows you to choose one of your project databases connections.

Security ?		
ATTRIBUTE	VALUE	DESCRIPTION
Use Security	<input type="checkbox"/>	Application needs validation to run. Read about the sc_apl_status macro.
Url output of the security	<input type="text"/>	Uri to be redirected if the user does not have access to this application.
Use Password	<input type="checkbox"/>	Enter password required to run the application.
Request password just once	<input type="checkbox"/>	Request password just once per session without asking for password every time you open the application in the same session.
Enable direct call by URL	<input checked="" type="checkbox"/>	Allow an application to be called by typing the URL directly in the browsers.

Application Security Configuration

Use Security

Turns the application accessible just when using the macro sc_apl_status or using the security module. When it is enabled, the application displays a message "User not Authorized" if the user tries to access it directly.

Url output of the security

It redirects the application to another URL after displays the warning "User not Authorized".

Use Password

Turns the application accessible just by using a password.

Request password just once

Request the password once per session, not needing to inform the password every time you access the application.

Enable direct call by URL

Allows to call the application directly from the URL in the browsers.

Allows calling the application directly from the URL in the browsers. In the development environment, you can disable those options for testing, it avoids requiring the login every time you run the application. To disable the security and password in development, you must access Options > My ScriptCase and uncheck the options: Enable Use of Security, Enable Use of Password.

This interface allows you to define a Log schema to the app. The Log schema tracks what the end-users are doing in the app. Those are the events it can tracks: Access, Insert, Update, and delete.

To learn how to create a Log schema [see here](#).

The screenshot shows a configuration window titled "LOG". It features a table with two columns: "ATTRIBUTE" and "VALUE". The "Schema" attribute has a dropdown menu currently set to "log". Below the table, there are two empty list boxes: "Events" on the left and "Access" on the right. Between these two boxes are two blue buttons: "Add >>" and "Remove <<".

Application Log configuration

Schema Select an existing log schema for the application.

Events You can select just specific events to save in the log.

Settings

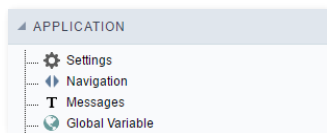
Global Variables

This screen shows all global variables used in the application

The global variable is an external parameter required for running the application. The application can include global variables in the WHERE clause, field definitions and names, event programming, and so on.

You can define global variables by using square brackets ([variable]). You can pass the parameters to the application through one of the methods: Session, Post, and Get

if you need to use [Database Identifiers](#) in or tables, we recommend using the double quotation marks(" ") instead of the brackets([]) because of the conflict with the [Global Variables syntax](#) of Scriptcase. Using the brackets as database identifiers may lead to problems in the Scriptcase applications.



Global variables Interface.

Example:

```
Select CustomerID, CustomerName from Customers WHERE CustomerID = '[v_customerid]'
```

We do not recommend using session variables (global variables) with the same name as the table fields.

In this case, the variable **v_customerid** is displaying in the global variables configuration.

ATTRIBUTE	VALUE
global	<div>Scope</div> <input type="checkbox"/> SESSION <input checked="" type="checkbox"/> POST <input checked="" type="checkbox"/> GET
	<div>Settings</div> <input type="checkbox"/> Optional
	<div>Type</div> <input type="radio"/> Out <input checked="" type="radio"/> In

Global variables configuration Interface.

Attribute

It shows the variable's name in the application.

Value

Defines the behavior of the variables, divided into three blocks, they are:

Scope

Defines how the application receives the variable. A variable defined as the POST method in the app does not receive value if it comes with the GET method. Those are the methods available:

- **ession**: Defines that the variable must be created as a PHP session variable by another application.
- **ET**: Defines that the variable must be passed by the URL, that is, visible by the browser.
- **OST**: Defines that the variable must be passed through the POST method of an HTML form.

Settings

- **ptional**: Check it to don't validate the variable at runtime.

Type (In / Out)

Defines whether the variable comes from another application (In), or it is starting in the current app (Out).

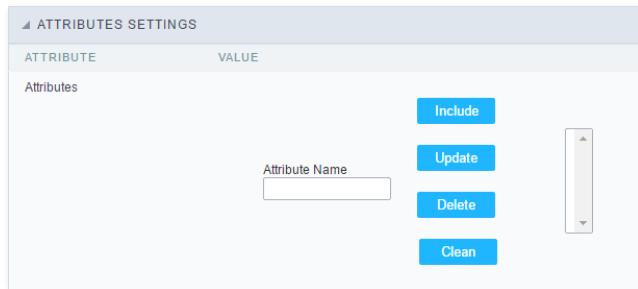
Description

Indicates where the application is using the variable.

ScriptCase has incorporated the concept of Object Oriented programming, using attributes, resources, methods and libraries. It is possible to create your own business rules in applications, and by using these concepts you can reap huge rewards in terms of better organization and improved development.

Attributes

The attributes are variables that has the global scope declared in the application. An attribute that can be referenced in all the methods and the application's events.

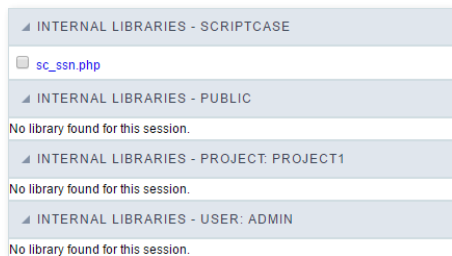


Attributes configuration Interface.

- **Attributes**
 - **Attribute Name** : This field allows to inform the attributes name, after typing the name just click on **include**. In case you want to **update** click on the attributes name (located on the buttons' right side), do the necessary modifications and click on update. In case it is necessary to **delete** an attribute, it is necessary to do same process as the update, but when selecting the attribute, it is necessary to click on the delete button. The **clear** button, as the name suggests, it clears the list of all the attributes listed on the right.

Internal libraries

To use the internal libraries just select the correspondent (Project Libraries for example). This process allows the methods existing in the library, to be visible in all the application's Events and Methods. See how to manage the libraries by [clicking here](#).

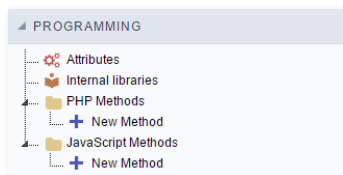


Internal Libraries management Interface

PHP Methods

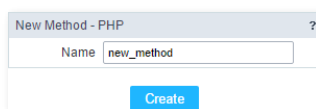
Methods are function or procedures declared by the developer, that helps when applying the routines. Using methods in the applications allows to reuse your code throughout the application, optimizing the development experience.

Creating A New Method



- Give the method a name and click Create. As in the image below.

PHP methods are incorporated into the same class, so it is not possible to use two functions with the same name even when used in different methods.



- Methods can receive parameters.

```
function new_method
1 echo "Hello World!!!";
```

- dd the amount of variables:

Definition of the parameters of the method:

new_method

- No defined parameter.

Add Parameter(s) Cancel

Save

- defining the variables:

Insertion of Parameters

Name	Type	Value Standard
<input type="text"/>	For Value	<input type="text"/>
	For Value	
	For References	

Save Back Cancel

- **ame** : Type in the variable's name.
- **ype** : Selecting the type of variables: For Value or For Reference.
- **alue Standard** : The parameter's value used to initialize when calling the method.

References allows to create a second name for a variable that you can use to read and modify the variable's original information.

- diting a parameter:

Definition of the parameters of the method:

new_method

Parameters

Stest = test

⏪ ⏩

☑ ☐ ✎ ✖

Add Parameter(s) Cancel

Save

- ☑ : Checking all the parameters in the parameter list.
- ☐ : Uncheck all the parameters in the parameter list.
- ✎ : Edit the selected parameter of the list.
- ✖ : Deletes the selected variable of the list.

New Application

The creation process of one application is simple enough and intuitive.

You just need to select which application you want to generate in the applications available in Scriptcase, That it will be generated automatically.

NEW APPLICATION

APPLICATION DATA RELATIONSHIP EDIT FIELDS THEME

Connection * Name *

conn_example

Table Fields

orders, order_details 23 selected

Localization *

English (United States) - UTF-8

SQL Select Statement *

```
SELECT
orders.orderid,
orders.customerid,
orders.employeeid,
orders.orderdate,
orders.requireddate,
orders.shippeddate,
orders.shipvia,
orders.freight,
orders.pricorder,
orders.shipcountry,
```

SQL Builder

- **onnection** - It defines the used connection to create the application. All the project's application will be displayed.
- **ame** - Application's name that is being created, It can not contain special characters.
- **ocalization** - Application's Language. The project's default language will be automatically selected.

Theme

In this screen you select the theme that the application will use. The default project's theme, that can be checked and changed in **Project > Properties**, will be selected automatically.

APPLICATION DATA RELATIONSHIP EDIT FIELDS THEME

Sc9_Rhino

Header

◀ ◀ ▶ ▶ Add Save

Block 1.1

Title 1 Object text

Block 2.1

Title 1	Title 2	Title 3
11111111	22222222	33333333
11111111	22222222	33333333

Footer

- **heme** - List of available themes in the project.

After all the settings are done, click in the button **Create**. In the next screen the settings of the created application will be displayed.

Desktop

The menu settings allow to define the attributes of the application display, according to the following:

Menu Settings		
ATTRIBUTE	VALUE	DESCRIPTION
Horizontal Alignment of the Menu	Left	Horizontal alignment mode of the menu.
Vertical Alignment of the items	Top	Vertical alignment mode of the menu items.
Horizontal Alignment of the Items	Left	Horizontal alignment mode of the menu items.
Toolbar Horizontal Alignment	Left	Alignment Mode of the toolbar items
Friendly URL		Friendly URL
Menu Width	100%	Table size width in pixels or percentage where the menu will be placed.
Menu Height		Table size height in pixels or percentage where the menu will be placed.
Iframe Width	100%	Iframe width in pixels or percentage where the application will be loaded.
Iframe Height	100%	Iframe height in pixels or percentage where the application will be loaded.
Hide menu items.	<input type="checkbox"/>	When security is being used, rather than disable the menu item, it hides the item. The item must have link to be able to be hidden.
Hide the menu's toolbar	<input type="checkbox"/>	When used with the security, instead of calling the toolbar item it will hide the item.
Use theme background	<input checked="" type="checkbox"/>	If set to Yes, will use the background theme of the menu, if set to No, will be used to set the background theme of the project.
Scroll bar	<input type="checkbox"/>	Allow scroll bar.
Select Language	<input type="checkbox"/>	Select the language on the bar menu.
Select Theme	<input type="checkbox"/>	Select theme on the top menu.
Alignment of Theme and Language	Right	Horizontal alignment of theme and language on the menu.
Always show the items	<input checked="" type="checkbox"/>	Show items even if the applications do not exist or if they were not generated.
Hide menu	<input type="checkbox"/>	Enables the option to hide the menu
Use expand/collapse	<input type="checkbox"/>	Enables an option that allows user to expand the application area, hiding menu items and header.

Menu configuration Interface.

Attributes

- Horizontal Alignment of the Menu**: Defines the horizontal alignment of the menu (Left, Center, Right).
- Vertical Alignment of the items**: Defines the vertical alignment of the menu items (Top, Center, Bottom).
- Horizontal Alignment of the Items**: Defines the horizontal alignment of the menu items (Left, Center, Right).
- Toolbar Horizontal Alignment**: Defines the horizontal alignment of the toolbar items (Left, Center, Right).
- Friendly URL**: Allows defining a name for the URL that's going to be called by the application. The characters allowed in the URL are(a-z, A-Z, 0-9). You still can inform the Friendly URL on the project home, where there is a list of applications with the "Friendly URL" column.
- Menu Width**: Define the width of the menu in pixels or percentages.
- Menu Height**: Define the height of the menu in pixels or percentages.
- Iframe Width**: Define the width of the Iframe (Where the applications will open in) in pixels or percentages.
- Iframe Height**: Define the height of the Iframe (Where the applications will open in) in pixels or percentages.
- Hide menu items**: When enabled and used with the security module, it'll hide the items that the user doesn't have permission to access, instead of only disabling them.
- Hide the menu's toolbar**: When enabled and used with the security module, it'll hide the toolbar that the user doesn't have permission to access, instead of only disabling them.
- Use theme background**: If enabled, it'll use the background theme of the menu. Else it'll use the background defined in the project.
- Scroll bar**: Enables the scroll bar in the Iframe.
- Select Language**: Allows to select the language in the menu bar.
- Select Theme**: Allows to select the theme in the menu bar.
- Alignment of Theme and Language**: Defines the horizontal alignment of the language and theme comboboxes.
- Always show the items**: Displays the application items, even though they don't exist, or haven't been generated yet.
- Hide menu**: Enables the option to hide the menu.

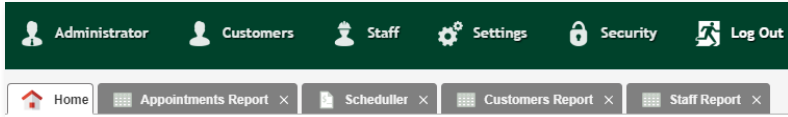
Navigation

The menu navigation settings allow to define the attributes of the tabs and the navigation path, according to the following:

Navigation		
ATTRIBUTE	VALUE	DESCRIPTION
Open items in a tab	<input checked="" type="checkbox"/>	Open the menu items in a different tab above the application
Tab context menu	<input checked="" type="checkbox"/>	Creates a context menu to manage the menu tabs
Menu tabs navigation	<input checked="" type="checkbox"/>	Activates the tab navigation when many tabs are opened at the same time.
Minimum tab width	<input type="text"/>	Minimum tab width to standardize the sizes of all the tabs
Maximum tab width	<input type="text"/>	Maximum width of the tab, if the width exceed this the text will be hidden
Use default icons in the tab	<input checked="" type="checkbox"/>	Uses the standard icons when the menu tab has no icon
Use default icons in the item	<input type="checkbox"/>	Uses the standard icons when the menu item has no icon
Show navigation breadcrumb	<input type="checkbox"/>	Show the navigation breadcrumb each time an opened application calls another application.
Default Application	Blank	Menu initial application.

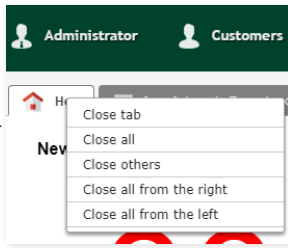
Menu Navigation configurations Interface.

- Attributes
 - Open items in a tab : When enabled, the items always open as tabs in a horizontal bar beneath the menu.



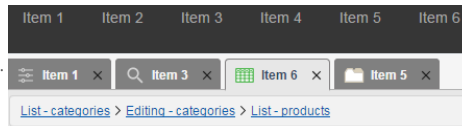
Navigating through Applications with tabs.

- Context menu of the Tabs: Allows you to manage the open tabs.



Application with a context menu.

- Minimum tab-width: Allows to inform the min-width of all the tabs.
- Maximum tab-width: Allows to inform the max-width of all the tabs.
- How navigation breadcrumb: Displays the path containing the application in the tab.



Navigation path Display in the Application.

- Use default icons in the item : When enabled, this option displays the standard icons for each item.
- Use default icons in the tab : When enabled, this option displays the standard icons for each tab.
- Default Application: Select a default application initiate when the menu opens.
- How the initial app as a tab : Allows to open the default application in a tab.

Mobile

Menu Settings		
ATTRIBUTE	VALUE	DESCRIPTION
Menu mobile	Normal	Define if the menu will be in the navigation mode or normal menu
Hide menu	<input checked="" type="checkbox"/>	Enables the option to hide the menu in mobile views
Menu's initial mode	Open	Informs the menu initial mode, if it is not set as visible
Float menu	<input type="checkbox"/>	Configures if the menu floats or occupies a physical space in the mobile view
Hide menu by clicking an item	<input checked="" type="checkbox"/>	Option to hide the menu after clicking an item, in case it is not set as visible

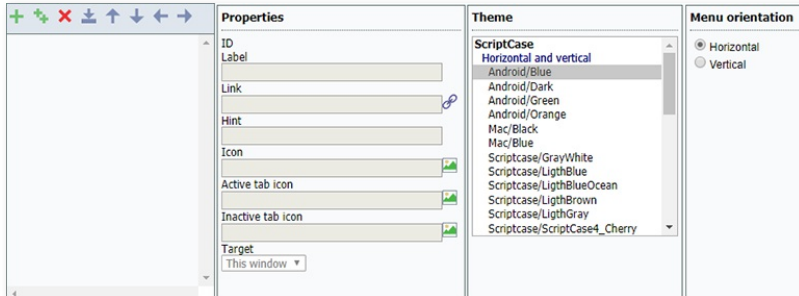
Mobile menu configuration Interface.

- Hide menu: Enables the option to hide the menu when viewing from a mobile device.
- Menu's initial mode: Inform the starter mode of the menu. Otherwise, it'll always be visible.
- Hide menu by clicking on an item : When enabled, it hides the menu when clicking on an item.
- Float Menu: Hides the icon to open the menu or always displays it.

Menu items settings

In this screen the user menu structure, indicating which applications will be called. Presents the following attributes, which should be completed by the user:

On mobile devices, if the menu item has sub-items and link, the link will be scrolled to display the sub-items.



Menu settings interface.

- **abel** : Title for the application within the menu. Type a fixed title or use the Scriptcase language system to create lang variable for these names, according to the languages used in the project.
- **ink** : Here you have to enter the application name that will be linked to the menu item. You can also click the search button and select from the application list. This link can also be an external URL.
- **int** : Descriptive text that helps the user to identify the application. It is an optional field.
- **con** : Image that will be displayed as an icon in the menu item. The user can click the search button to select the image from the Scriptcase Image Manage. It is an optional field.
- **ctive tab icon** : Image that will be displayed as an icon in the menu item when active. The user can click the search button to select the image from the Scriptcase Image Manage. It is an optional field. If you leave it blank Scriptcase adds a default image.
- **nactive tab icon** : Image that will be displayed as an icon in the menu item when inactive. The user can click the search button to select the image from the Scriptcase Image Manage. It is an optional field. If you leave it blank Scriptcase adds a default image.
- **arget** : This option sets how the application will be displayed, it can be targeted as: Same window, Another window or leave (it will log the user out from the system).

Resources for menu items creation

- **nsert item** : This option add items to the menu.
- **nsert Sub-Item** : This option add sub-items to the menu, you might select the main item before to add the sub-items within the selected main item. You can also manage the sub-item position using the arrows.
- **emove** : This option deletes the item/sub-item, you can select the items before and then remove.
- **mport applications** : You can use this option to call an application to attach to the item. It will open the project application list.
- **ove up** : This option allows you to move up a menu item or sub-item.
- **ove down** : This option allows you to move down a menu item or sub-item.
- **ove left** : This option allows you to move a menu item or sub-item to the left. You can use it to organize the menu hierarchy.
- **ove right** : This option allows you to move a menu item or sub-item to the right. You can use it to organize the menu hierarchy.

Importing the Applications

Importing applications is easy and fast. You can select to import one or more applications from a list. Then you need to organize the menu levels as you wish, by using the positioning buttons.

samples			
Application	Directory	Dt. Creating	Description
calendar01	root/Calendar	14/07/10 17:15	Calendar Application *
calendar02	root/Calendar	19/07/10 18:43	
calendar03	root/Calendar	29/10/10 17:32	Calendar per user
calendar_color_event	root/Calendar	25/07/14 00:11	Calendar Application with print button and event color per day *
chart01	root/Grid_Applications/Charts	25/01/13 10:02	Bar HTML 5 Chart *

Applications list to Import.

The link property receives the application name, as the image below.

Menu Settings

Properties

ID (item_185)

Label
{lang_header_form_64_title}

Link
form_64

Hint
form_64

Icon

Active tab icon

Inactive tab icon

Target
This window

Theme

ScriptCase

Horizontal and vertical

- Android/Blue
- Android/Dark
- Android/Green
- Android/Orange
- Mac/Black
- Mac/Blue
- Scriptcase/GrayWhite
- Scriptcase/LigthBlue
- Scriptcase/LigthBlueOcean
- Scriptcase/LigthBrown
- Scriptcase/LigthGray
- Scriptcase/ScriptCase4_Cherry

Menu orientation

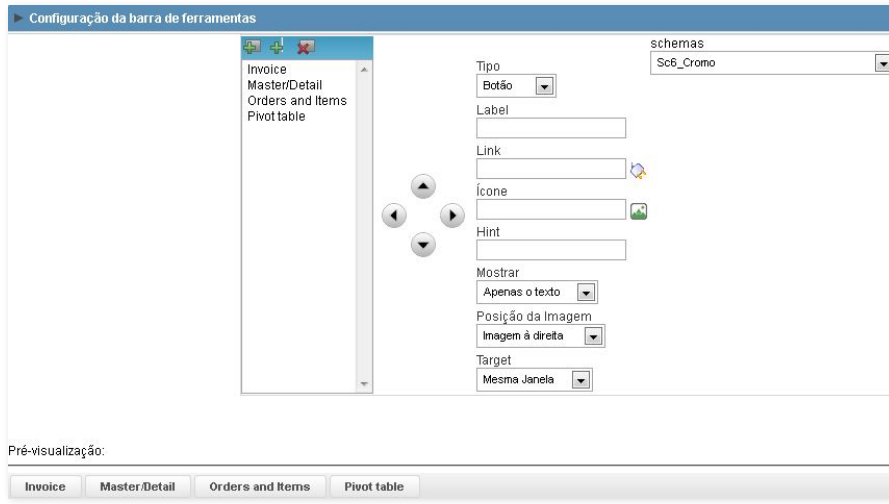
Horizontal

Vertical

Application Import Interface.

Settings

In this interface, you can build a shortcut bar in the Menu application that enables the connection between applications through buttons, icons, and text links.



Items creation (Icons in the menu list top):

Insert Item	Add a new item in the menu toolbar.
Separator Insert	Add a separator in the toolbar.
Remove	Removes a menu item. Select the item or sub-item and click on the button to remove it.

Moving items (Arrows):

Move Up	Moves an item or sub-item from the toolbar up.
Move Down	Moves down an item or sub-item from the toolbar.
Move Left	Moves an item or sub-item from the toolbar to the left.
Move Right	Moves an item or sub-item from the toolbar to the left.

Settings Items:

Type	Allows inserting a link, a button, or an icon.
Label	Allows defining the application title displayed in the menu.
Link	Allows selecting the application through his name or by a link.
Icon	Allows selecting an image to display as an icon.
Hint	Allows typing a description text to help the identification of the application by the user.
Display	This option allows choosing if the item shows only texts, only images, or both.
Image Position	Allows defining if the image displays on the right or left side.
Target	Allows defining how to display the applications: In the same window or in a new one.

Security ?		
ATTRIBUTE	VALUE	DESCRIPTION
Use Security	<input type="checkbox"/>	Application needs validation to run. Read about the sc_apl_status macro.
Url output of the security	<input type="text"/>	Uri to be redirected if the user does not have access to this application.
Use Password	<input type="checkbox"/>	Enter password required to run the application.
Request password just once	<input type="checkbox"/>	Request password just once per session without asking for password every time you open the application in the same session.
Enable direct call by URL	<input checked="" type="checkbox"/>	Allow an application to be called by typing the URL directly in the browsers.

Application Security Configuration

Use Security

Turns the application accessible just when using the macro sc_apl_status or using the security module. When it is enabled, the application displays a message "User not Authorized" if the user tries to access it directly.

Url output of the security

It redirects the application to another URL after displays the warning "User not Authorized".

Use Password

Turns the application accessible just by using a password.

Request password just once

Request the password once per session, not needing to inform the password every time you access the application.

Enable direct call by URL

Allows to call the application directly from the URL in the browsers.

Allows calling the application directly from the URL in the browsers. In the development environment, you can disable those options for testing, it avoids requiring the login every time you run the application. To disable the security and password in development, you must access Options > My ScriptCase and uncheck the options: Enable Use of Security, Enable Use of Password.

This interface allows you to define a Log schema to the app. The Log schema tracks what the end-users are doing in the app. Those are the events it can tracks: Access, Insert, Update, and delete.

To learn how to create a Log schema [see here](#).

The screenshot shows a configuration window titled "LOG". It features a table with two columns: "ATTRIBUTE" and "VALUE". The "Schema" attribute has a dropdown menu currently set to "log". Below the table, there are two empty list boxes: "Events" on the left and "Access" on the right. Between these two boxes are two blue buttons: "Add >>" and "Remove <<".

Application Log configuration

Schema Select an existing log schema for the application.

Events You can select just specific events to save in the log.

Within the events you can include Scriptcase global, local variables, libraries and macros, PHP, JavaScript, HTML and CSS codes.

onApplicationInit

This event occurs only once, when the application is loaded and before the application executes the SQL. It can be used for data processing or checking variables.

Example: `if ([glo_var_dept] != 'financial'){ sc_redir(app_x.php); }`

onExecute

This event runs when the item of the menu is clicked or an application is selected.

onLoad

This event runs when the application is finished loading.

This module lets you editing display options for the application. You can apply visualization themes, organize blocks, define values, and the display format of the Header and Footer.

Layout Settings

ScriptCase creates an app by using default values defined per project, but you can change those values for a specific application on this interface.

LAYOUT SETTINGS

Header Template: **Flat** (Template name used for the application header.)

Footer Template: **Default** (Template name used for the application footer.)

Button: (empty) (Use different buttons than what was defined in the color scheme.)

Themes: **Sc9_Rhino** (Use different themes from the one defined for the color scheme)

Header

xxyyzz xxxxx yyyyy

Block 1

Name: xxxxxxxxxxx

Type: Male Female

Address*: xxxxxxxxxxx

Groups*: Male Female

Countries: Afghanistan

Address: yyyyyyyyyyyyyyy

Photos: Drag & Drop files here

Image1.png ✓
Image2.png ✗

Captcha:

Application Layout Settings

Header Template

Allows choosing the look of the application Header.

Footer Template

Allows choosing the look of the application Footer.

Button

Allows choosing the button theme for the application.

Themes

Choose one of the selected themes in the project properties. It defines the look of the application, like colors, fonts, and others.

Header & Footer

Header

Here we can define if we want to display the header or not, the application title, and fill the header variables. The header looks according to the selected template in the Layout settings interface.

HEADER	
ATTRIBUTE	VALUE
Display Header	<input checked="" type="checkbox"/>
Insertion Title	<input type="text" value="{lang_othr_frmj_tit} - orders"/>
Update Title	<input type="text" value="{lang_othr_frmu_tit} - orders"/>
Header Variables	
SC_HEADER	<input type="text" value="Title"/>
SC_VALUE	<input type="text" value="Date"/> ?

Application Header Configuration

Display Header


This option determines if we want to display the header or not.

Title

Allows informing the title of the application.

Header Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **itle**: It displays the value of "**Application Title**" in the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Depending on the Application, you may have more than one title option.

Footer

FOOTER	
ATTRIBUTE	VALUE
Display Footer	<input type="checkbox"/>
Footer Variables	
NM_FOOTER_DATA	<input type="text"/>
NM_FOOTER_LOGO	<input type="text"/>
NM_FOOTER_TITLE	<input type="text"/>

Application Footer Settings


This page may change depending on the footer template chosen in the Layout Settings.

Display Footer

This option determines if we want to display the footer or not.

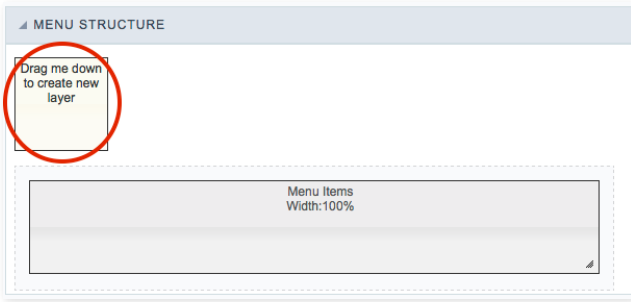
Footer Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Menu Structure

Using this option you can change the entire menu structure.



Using this option, you can change the entire menu structure. Drag the selected option to the “Menu Items” area to create a new “Layer” where you can add the contents. You can increase or decrease the layer width and also change its positioning within the “Menu items” area.

After adding the layer and customize its size, you can click on the “pencil” icon to edit the layer options or click on the “trash” icon to delete it.

Layer Editing

Name	<input type="text" value="Layer 0"/>	Display	<input checked="" type="checkbox"/>
Width	<input type="text" value="6.13%"/>	Alignment	<input type="text" value="Left"/>
Type	Font	Background color	Font color
<input type="text" value="↑"/>	<input type="text" value="16"/> B <i>I</i>	<input type="text" value="#000000"/>	<input type="text" value=""/>
<input type="button" value="Add"/>	<input type="button" value="Close"/> <input type="button" value="Save changes"/>		

Name

Use this option to set a name for the layer.

Width

It changes the layer size by using percentages or pixels. You need to change the “Menu Width” option within the Menu “Settings”. This value is in percentage as default.

Display

This option sets whether to display the layer or not.

Alignment

Using this option, you can change the alignment of all layer contents in the Left, Right, or Center.

Type

Using this option, you can select the data type of each row of the layer.

***Title**: It displays the application title, according to the value set on “Application Title” within the menu “Header & Footer”.

***Date**: It displays a date using your preferred date format. Example: d-m-Y

***Image**: Add an image to the menu. Use the link to access the Image Manager from Scriptcase to choose an image or upload a new one.

***Value**: It displays the content of the text input. You can inform static texts and “ **Global Variables**” such as the logged user, for example: [usr_login].

***Library**: Select an External Library from Scriptcase. You can import or create new libraries using the option “Tools » External Libraries” from the Scriptcase main menu.

***Method**: Select an application method to apply to the layer. Create new methods by using the option “PHP Programming” within the menu “Programming.”

Font

Select one of the available font families from the list. Change the font size and set the text as bold and/or italic.

Background color

Change the background color of each row of the layer content.

Font color

Change the color for the text in the layer.

Delete

It allows deleting the layer line.

Add

It allows adding another line to the layer.

Close

It closes the edit screen of the layer.

Save

Save all changes for the layer.

Settings

Convert menu type

Convert Menu to Tree Menu

Would you like to convert your **Menu** application to **Tree Menu**?

Menu conversion Interface.

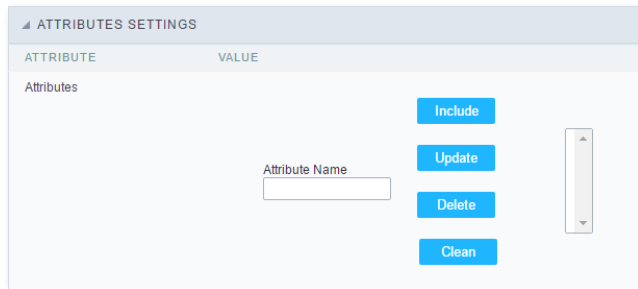
Convert the Menu to a Tree Menu or vice versa.

*It doesn't change any of the other original settings, except the format.

ScriptCase has incorporated the concept of Object Oriented programming, using attributes, resources, methods and libraries. It is possible to create your own business rules in applications, and by using these concepts you can reap huge rewards in terms of better organization and improved development.

Attributes

The attributes are variables that has the global scope declared in the application. An attribute that can be referenced in all the methods and the application's events.



Attributes configuration Interface.

- **Attributes**
 - **Attribute Name** : This field allows to inform the attributes name, after typing the name just click on **include**. In case you want to **update** click on the attributes name (located on the buttons' right side), do the necessary modifications and click on update. In case it is necessary to **delete** an attribute, it is necessary to do same process as the update, but when selecting the attribute, it is necessary to click on the delete button. The **clear** button, as the name suggests, it clears the list of all the attributes listed on the right.

Internal libraries

To use the internal libraries just select the correspondent (Project Libraries for example). This process allows the methods existing in the library, to be visible in all the application's Events and Methods. See how to manage the libraries by [clicking here](#).

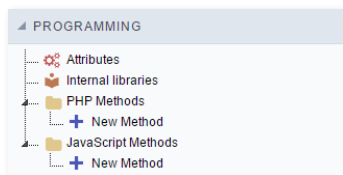


Internal Libraries management Interface

PHP Methods

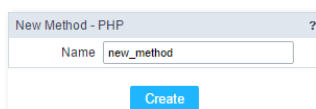
Methods are function or procedures declared by the developer, that helps when applying the routines. Using methods in the applications allows to reuse your code throughout the application, optimizing the development experience.

Creating A New Method



- Give the method a name and click Create. As in the image below.

PHP methods are incorporated into the same class, so it is not possible to use two functions with the same name even when used in different methods.



- Methods can receive parameters.

```
function new_method
1 echo "Hello World!!!";
```

- dd the amount of variables:

Definition of the parameters of the method:

new_method

- No defined parameter.

Add Parameter(s) Cancel

Save

- defining the variables:

Insertion of Parameters

Name	Type	Value Standard
<input type="text"/>	For Value	<input type="text"/>
	For Value	
	For References	

Save Back Cancel

- **ame** : Type in the variable's name.
- **ype** : Selecting the type of variables: For Value or For Reference.
- **alue Standard** : The parameter's value used to initialize when calling the method.

References allows to create a second name for a variable that you can use to read and modify the variable's original information.

- diting a parameter:

Definition of the parameters of the method:

new_method

Parameters

Stest = test

⏪ ⏩

☑ ☐ ✎ ✖

Add Parameter(s) Cancel

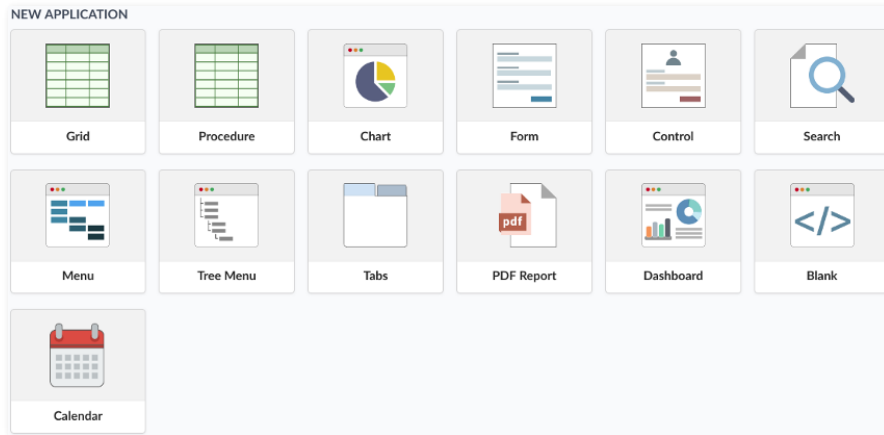
Save

- ☑ : Checking all the parameters in the parameter list.
- ☐ : Uncheck all the parameters in the parameter list.
- ✎ : Edit the selected parameter of the list.
- ✖ : Deletes the selected variable of the list.

New Application

The creation process of one application is simple enough and intuitive.

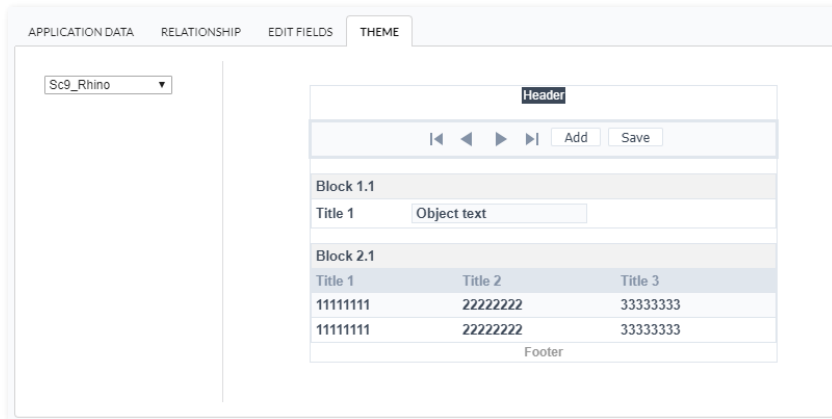
You just need to select which application you want to generate in the applications available in Scriptcase, That it will be generated automatically.



- **ame** - Application's name that is being created, It can not contain special characters.
- **ocalization** - Application's Language. The project's default language will be automatically selected.

Theme

In this screen you select the theme that the application will use. The default project's theme, that can be checked and changed in **Project > Properties**, will be selected automatically.

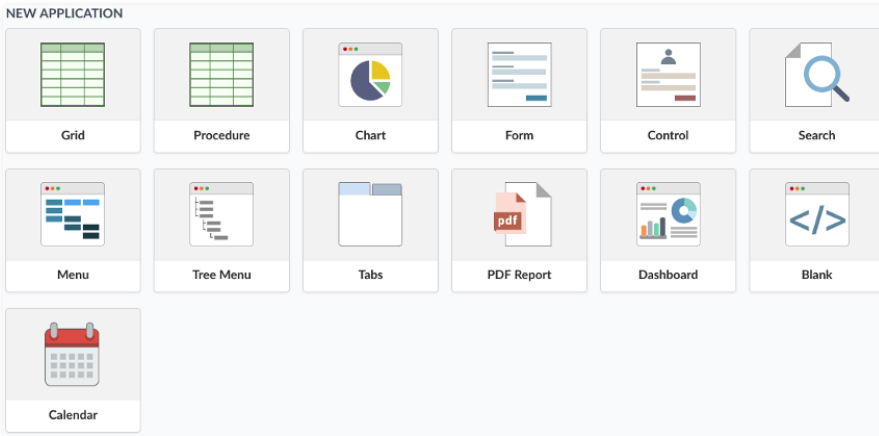


- **heme** - List of available themes in the project.

*After all the settings are done, click in the button **Create**. In the next screen the settings of the created application will be displayed.*

The creation process of one application is simple enough and intuitive.

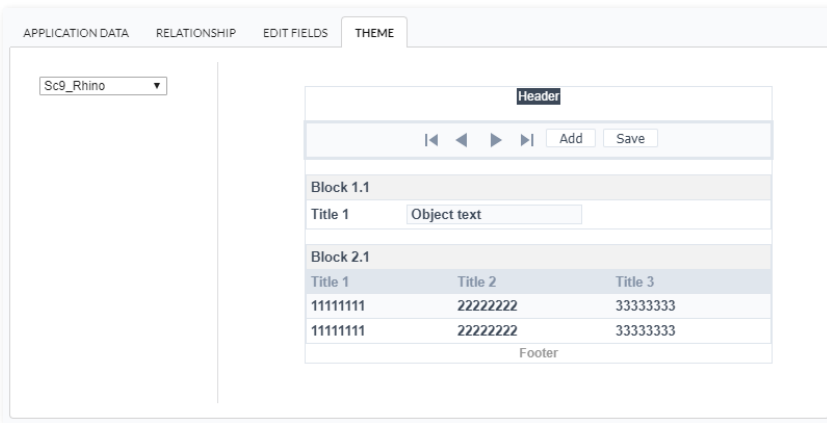
You just need to select which application you want to generate in the applications available in Scriptcase, That it will be generated automatically.



- **ame** - Application's name that is being created, It can not contain special characters.
- **ocalization** - Application's Language. The project's default language will be automatically selected.

Theme

In this screen you select the theme that the application will use. The default project's theme, that can be checked and changed in **Project > Properties**, will be selected automatically.



- **heme** - List of available themes in the project.

After all the settings are done, click in the button **Create**. In the next screen the settings of the created application will be displayed.

Desktop

The menu settings allow to define the attributes of the application display, according to the following:

Menu Settings ?		
ATTRIBUTE	VALUE	DESCRIPTION
Horizontal Alignment of the Menu	<input type="text" value="Left"/>	Horizontal alignment mode of the menu.
Vertical Alignment of the items	<input type="text" value="Top"/>	Vertical alignment mode of the menu items.
Horizontal Alignment of the Items	<input type="text" value="Left"/>	Horizontal alignment mode of the menu items.
Toolbar Horizontal Alignment	<input type="text" value="Left"/>	Alignment Mode of the toolbar items
Friendly URL	<input type="text"/>	Friendly URL
Menu Width	<input type="text" value="100%"/>	Table size width in pixels or percentage where the menu will be placed.
Menu Height	<input type="text"/>	Table size height in pixels or percentage where the menu will be placed.
Iframe Width	<input type="text" value="100%"/>	Iframe width in pixels or percentage where the application will be loaded.
Iframe Height	<input type="text" value="100%"/>	Iframe height in pixels or percentage where the application will be loaded.
Hide menu items.	<input type="checkbox"/>	When security is being used, rather than disable the menu item, it hides the item. The item must have link to be able to be hidden.
Hide the menu's toolbar	<input type="checkbox"/>	When used with the security, instead of calling the toolbar item it will hide the item.
Use theme background	<input checked="" type="checkbox"/>	If set to Yes, will use the background theme of the menu, if set to No, will be used to set the background theme of the project.
Scroll bar	<input type="checkbox"/>	Allow scroll bar.
Select Language	<input type="checkbox"/>	Select the language on the bar menu.
Select Theme	<input type="checkbox"/>	Select theme on the top menu.
Alignment of Theme and Language	<input type="text" value="Right"/>	Horizontal alignment of theme and language on the menu.
Always show the items	<input checked="" type="checkbox"/>	Show items even if the applications do not exist or if they were not generated.
Hide menu	<input type="checkbox"/>	Enables the option to hide the menu
Use expand/collapse	<input type="checkbox"/>	Enables an option that allows user to expand the application area, hiding menu items and header.

Menu configuration Interface.

- **Attributes**
 - **Horizontal Alignment of the Menu**: Defines the horizontal alignment of the menu (Left, Center, Right).
 - **Vertical Alignment of the items**: Defines the vertical alignment of the menu items (Top, Center, Bottom).
 - **Horizontal Alignment of the Items**: Defines the horizontal alignment of the menu items (Left, Center, Right).
 - **Toolbar Horizontal Alignment**: Defines the horizontal alignment of the toolbar items (Left, Center, Right).
 - **Friendly URL**: Allows defining a name for the URL that's going to be called by the application. The characters allowed in the URL are(a-z, A-Z, 0-9). You still can inform the Friendly URL on the project home, where there is a list of applications with the "Friendly URL" column.
 - **Menu Width**: Define the width of the menu in pixels or percentages.
 - **Menu Height**: Define the height of the menu in pixels or percentages.
 - **Iframe Width**: Define the width of the Iframe (Where the applications will open in) in pixels or percentages.
 - **Iframe Height**: Define the height of the Iframe (Where the applications will open in) in pixels or percentages.
 - **Hide menu items**: When enabled and used with the security module, it'll hide the items that the user doesn't have permission to access, instead of only disabling them.
 - **Hide the menu's toolbar**: When enabled and used with the security module, it'll hide the toolbar that the user doesn't have permission to access, instead of only disabling them.
 - **Use theme background**: If enabled, it'll use the background theme of the menu. Else it'll use the background defined in the project.
 - **Scroll bar**: Enables the scroll bar in the Iframe.
 - **Select Language**: Allows to select the language in the menu bar.
 - **Select Theme**: Allows to select the theme in the menu bar.
 - **Alignment of Theme and Language**: Defines the horizontal alignment of the language and theme comboboxes.
 - **Always show the items**: Displays the application items, even though they don't exist, or haven't been generated yet.
 - **Hide menu**: Enables the option to hide the menu.

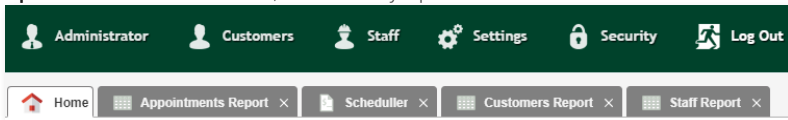
Navigation

The menu navigation settings allow to define the attributes of the tabs and the navigation path, according to the following:

Navigation		
ATTRIBUTE	VALUE	DESCRIPTION
Open items in a tab	<input checked="" type="checkbox"/>	Open the menu items in a different tab above the application
Tab context menu	<input checked="" type="checkbox"/>	Creates a context menu to manage the menu tabs
Menu tabs navigation	<input checked="" type="checkbox"/>	Activates the tab navigation when many tabs are opened at the same time.
Minimum tab width	<input type="text"/>	Minimum tab width to standardize the sizes of all the tabs
Maximum tab width	<input type="text"/>	Maximum width of the tab, if the width exceed this the text will be hidden
Use default icons in the tab	<input checked="" type="checkbox"/>	Uses the standard icons when the menu tab has no icon
Use default icons in the item	<input type="checkbox"/>	Uses the standard icons when the menu item has no icon
Show navigation breadcrumb	<input type="checkbox"/>	Show the navigation breadcrumb each time an opened application calls another application.
Default Application	Blank	Menu initial application.

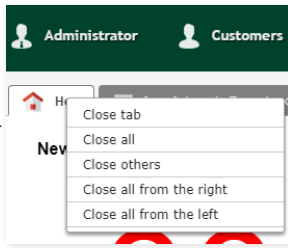
Menu Navigation configurations Interface.

- Attributes
 - Open items in a tab : When enabled, the items always open as tabs in a horizontal bar beneath the menu.



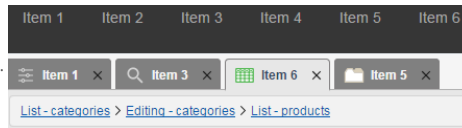
Navigating through Applications with tabs.

- Context menu of the Tabs: Allows you to manage the open tabs.



Application with a context menu.

- Minimum tab-width: Allows to inform the min-width of all the tabs.
- Maximum tab-width: Allows to inform the max-width of all the tabs.
- How navigation breadcrumb: Displays the path containing the application in the tab.



Navigation path Display in the Application.

- Use default icons in the item : When enabled, this option displays the standard icons for each item.
- Use default icons in the tab : When enabled, this option displays the standard icons for each tab.
- Default Application: Select a default application initiate when the menu opens.
- How the initial app as a tab : Allows to open the default application in a tab.

Mobile

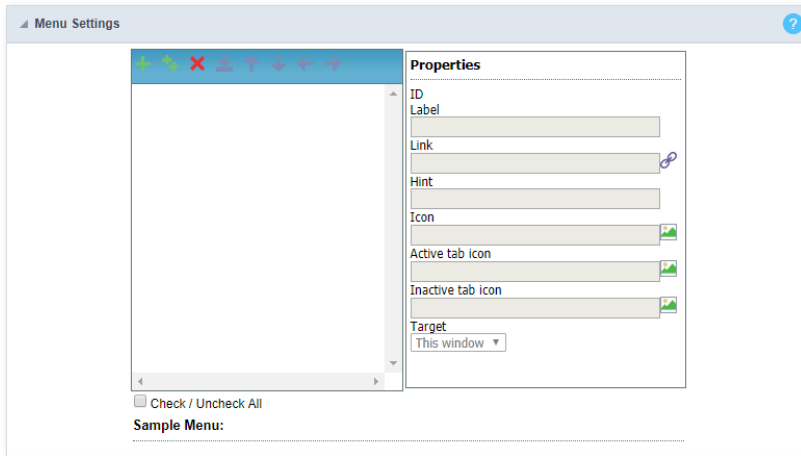
Menu Settings		
ATTRIBUTE	VALUE	DESCRIPTION
Menu mobile	Normal	Define if the menu will be in the navigation mode or normal menu
Hide menu	<input checked="" type="checkbox"/>	Enables the option to hide the menu in mobile views
Menu's initial mode	Open	Informs the menu initial mode, if it is not set as visible
Float menu	<input type="checkbox"/>	Configures if the menu floats or occupies a physical space in the mobile view
Hide menu by clicking an item	<input checked="" type="checkbox"/>	Option to hide the menu after clicking an item, in case it is not set as visible

Mobile menu configuration Interface.

- Hide menu: Enables the option to hide the menu when viewing from a mobile device.
- Menu's initial mode: Inform the starter mode of the menu. Otherwise, it'll always be visible.
- Hide menu by clicking on an item : When enabled, it hides the menu when clicking on an item.
- Float Menu: Hides the icon to open the menu or always displays it.

Menu items settings

In this screen the user menu structure, indicating which applications will be called. Presents the following attributes, which should be completed by the user:



Tree menu settings interface.

- **abel** : Title for the application within the menu. You type a fixed title or use the Scriptcase language system to create lang variable for these names, according to the languages used in the project.
- **ink** : Here you have to enter the application name that will be linked to the menu item. You can also click search button and select from the application list. This link can also be an external URL.
- **int** : Descriptive text that helps the user to identify the application. It is an optional field.
- **con** : Image that will be displayed as an icon in the menu item. The user can click the search button to select the image from the Scriptcase Image Manage. It is an optional field.
- **ctive tab icon** : Image that will be displayed as an icon in the menu item when active. The user can click the search button to select the image from the Scriptcase Image Manage. It is an optional field. If you leave it blank Scriptcase adds a default image.
- **nactive tab icon** : Image that will be displayed as an icon in the menu item when inactive. The user can click the search button to select the image from the Scriptcase Image Manage. It is an optional field. If you leave it blank Scriptcase adds a default image.
- **arget** : This option sets how the application will be displayed, it can be target as: Same window, Another window or leave (it will log the user out from the system).

Resources for menu items creation

- **nsert item** : This option add items to the menu
- **nsert Sub-Item** : This option add sub-items to the menu, you might select the main item before to add the sub-items within the selected main item. You can also manage the sub-item position using the arrows.
- **emove** : This option deletes the item/sub-item, you might select the items before and then remove.
- **mport applications** : You can use this option to call an application to attach to the item. It will open the project application list.
- **ove up** : This option allows you to move up a menu item or sub-item.
- **ove down** : This option allows you to move down a menu item or sub-item.
- **ove left** : This option allows you to move a menu item or sub-item to the left. You can use it to organize the menu hierarchy.
- **ove right** : This option allows you to move a menu item or sub-item to the right. You can use it to organize the menu hierarchy.

Importing the Applications

Importing applications is easy and fast. You can select to import one or more applications from a list. Then you need to organize the menu levels as you wish, by using the positioning buttons.

Application	Directory	Dt. Creating	Description
calendar01	root/Calendar	14/07/10 17:15	Calendar Application *
calendar02	root/Calendar	19/07/10 18:43	
calendar03	root/Calendar	29/10/10 17:32	Calendar per user
calendar_color_event	root/Calendar	25/07/14 00:11	Calendar Application with print button and event color per day *
chart01	root/Grid_Applications/Charts	25/01/13 10:02	Bar HTML 5 Chart *

Applications list to Import.

The link property receives the application name, as the image below.

Menu Settings

Properties

ID (item_185)
Label
{lang_header_form_64_title}
Link
form_64
Hint
form_64
Icon
Active tab icon
Inactive tab icon
Target
This window

Theme

ScriptCase
Horizontal and vertical
Android/Blue
Android/Dark
Android/Green
Android/Orange
Mac/Black
Mac/Blue
Scriptcase/GrayWhite
Scriptcase/LigthBlue
Scriptcase/LigthBlueOcean
Scriptcase/LigthBrown
Scriptcase/LigthGray
Scriptcase/ScriptCase4_Cherry

Menu orientation

Horizontal
 Vertical

Application Import Interface.

Security ?		
ATTRIBUTE	VALUE	DESCRIPTION
Use Security	<input type="checkbox"/>	Application needs validation to run. Read about the sc_apl_status macro.
Url output of the security	<input type="text"/>	Uri to be redirected if the user does not have access to this application.
Use Password	<input type="checkbox"/>	Enter password required to run the application.
Request password just once	<input type="checkbox"/>	Request password just once per session without asking for password every time you open the application in the same session.
Enable direct call by URL	<input checked="" type="checkbox"/>	Allow an application to be called by typing the URL directly in the browsers.

Application Security Configuration

Use Security

Turns the application accessible just when using the macro sc_apl_status or using the security module. When it is enabled, the application displays a message "User not Authorized" if the user tries to access it directly.

Url output of the security

It redirects the application to another URL after displays the warning "User not Authorized".

Use Password

Turns the application accessible just by using a password.

Request password just once

Request the password once per session, not needing to inform the password every time you access the application.

Enable direct call by URL

Allows to call the application directly from the URL in the browsers.

Allows calling the application directly from the URL in the browsers. In the development environment, you can disable those options for testing, it avoids requiring the login every time you run the application. To disable the security and password in development, you must access Options > My ScriptCase and uncheck the options: Enable Use of Security, Enable Use of Password.

This interface allows you to define a Log schema to the app. The Log scheme tracks what the end-users are doing in the app. Those are the events it can tracks: Access, Insert, Update, and delete.

To learn how to create a Log schema [see here](#).

The screenshot shows a configuration window titled "LOG". It features a table with two columns: "ATTRIBUTE" and "VALUE". The "Schema" attribute has a dropdown menu currently set to "log". Below the table, there are two empty list boxes: "Events" on the left and "Access" on the right. Between these two boxes are two blue buttons: "Add >>" and "Remove <<".

Application Log configuration

Schema Select an existing log schema for the application.

Events You can select just specific events to save in the log.

Within the events you can include Scriptcase global, local variables, libraries and macros, PHP, JavaScript, HTML and CSS codes.

onApplicationInit

This event occurs only once, when the application is loaded and before the application executes the SQL. It can be used for data processing or checking variables.

Example: `if ([glo_var_dept] != 'financial'){ sc_redir(app_x.php); }`

onExecute

This event runs when the item of the menu is clicked or an application is selected.

onLoad

This event runs when the application is finished loading.

This module lets you editing display options for the application. You can apply visualization themes, organize blocks, define values, and the display format of the Header and Footer.

Layout Settings

ScriptCase creates an app by using default values defined per project, but you can change those values for a specific application on this interface.

LAYOUT SETTINGS

Header Template	<input type="text" value="Flat"/>	Template name used for the application header.
Footer Template	<input type="text" value="Default"/>	Template name used for the application footer.
Button	<input type="text"/>	Use different buttons than what was defined in the color scheme.
Themes	<input type="text" value="Sc9_Rhino"/>	Use different themes from the one defined for the color scheme

Header

xyyyzz
xxxxx
yyyyy ▾

Block 1

Name

Type
 Male
 Female

Address*

Groups*
 Male
 Female


Countries

Address

Photos

Drag & Drop files here

Image1.png ✓
Image2.png ✗

Captcha 

Application Layout Settings

Header Template

Allows choosing the look of the application Header.

Footer Template

Allows choosing the look of the application Footer.

Button

Allows choosing the button theme for the application.

Themes

Choose one of the selected themes in the project properties. It defines the look of the application, like colors, fonts, and others.

Header & Footer

Header

Here we can define if we want to display the header or not, the application title, and fill the header variables. The header looks according to the selected template in the Layout settings interface.

HEADER	
ATTRIBUTE	VALUE
Display Header	<input checked="" type="checkbox"/>
Insertion Title	{lang_othr_frmj_tit} - orders
Update Title	{lang_othr_frmu_tit} - orders
Header Variables	
SC_HEADER	Title
SC_VALUE	Date

Application Header Configuration

Display Header


This option determines if we want to display the header or not.

Title

Allows informing the title of the application.

Header Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **itle**: It displays the value of "**Application Title**" in the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Depending on the Application, you may have more than one title option.

Footer

FOOTER	
ATTRIBUTE	VALUE
Display Footer	<input type="checkbox"/>
Footer Variables	
NM_FOOTER_DATA	
NM_FOOTER_LOGO	
NM_FOOTER_TITLE	

Application Footer Settings


This page may change depending on the footer template chosen in the Layout Settings.

Display Footer

This option determines if we want to display the footer or not.

Footer Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Settings

Convert menu type

Convert Menu to Tree Menu

Would you like to convert your **Menu** application to **Tree Menu**?

Menu conversion Interface.

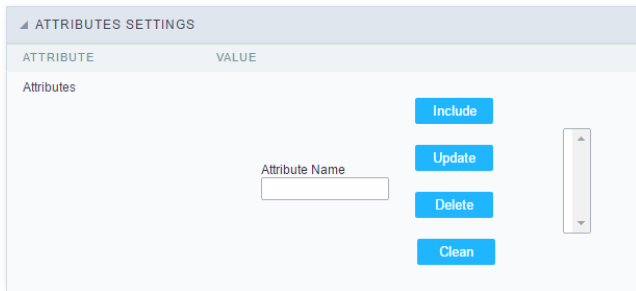
Convert the Menu to a Tree Menu or vice versa.

*It doesn't change any of the other original settings, except the format.

ScriptCase has incorporated the concept of Object Oriented programming, using attributes, resources, methods and libraries. It is possible to create your own business rules in applications, and by using these concepts you can reap huge rewards in terms of better organization and improved development.

Attributes

The attributes are variables that has the global scope declared in the application. An attribute that can be referenced in all the methods and the application's events.



Attributes configuration Interface.

- **attributes**
 - **attribute Name** : This field allows to inform the attributes name, after typing the name just click on **include**. In case you want to **update** click on the attributes name (located on the buttons' right side), do the necessary modifications and click on update. In case it is necessary to **delete** an attribute, it is necessary to do same process as the update, but when selecting the attribute, it is necessary to click on the delete button. The **clear** button, as the name suggests, it clears the list of all the attributes listed on the right.

Internal libraries

To use the internal libraries just select the correspondent (Project Libraries for example). This process allows the methods existing in the library, to be visible in all the application's Events and Methods. See how to manage the libraries by [clicking here](#).



Internal Libraries management Interface

PHP Methods

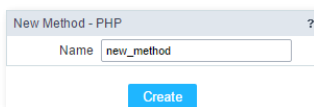
Methods are function or procedures declared by the developer, that helps when applying the routines. Using methods in the applications allows to reuse your code throughout the application, optimizing the development experience.

Creating A New Method



- Give the method a name and click Create. As in the image below.

PHP methods are incorporated into the same class, so it is not possible to use two functions with the same name even when used in different methods.



- Methods can receive parameters.

```
function new_method
1 echo "Hello World!!!";
```

- dd the amount of variables:

Definition of the parameters of the method:

new_method

- No defined parameter.

Add 1 Parameter(s) Cancel

Save

- defining the variables:

Insertion of Parameters

Name	Type	Value Standard
	For Value	
	For Value	
	For References	

Save Back Cancel

- **ame** : Type in the variable's name.
- **ype** : Selecting the type of variables: For Value or For Reference.
- **alue Standard** : The parameter's value used to initialize when calling the method.

References allows to create a second name for a variable that you can use to read and modify the variable's original information.

- diting a parameter:

Definition of the parameters of the method:

new_method

Parameters
Stest = test

☑ ☐ ✎ ✕

Add 1 Parameter(s) Cancel

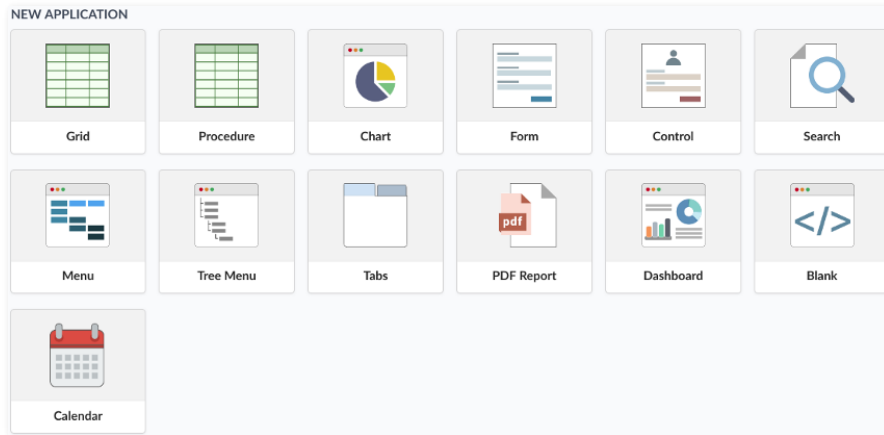
Save

- ☑ : Checking all the parameters in the parameter list.
- ☐ : Uncheck all the parameters in the parameter list.
- ✎ : Edit the selected parameter of the list.
- ✕ : Deletes the selected variable of the list.

New Application

The creation process of one application is simple enough and intuitive.

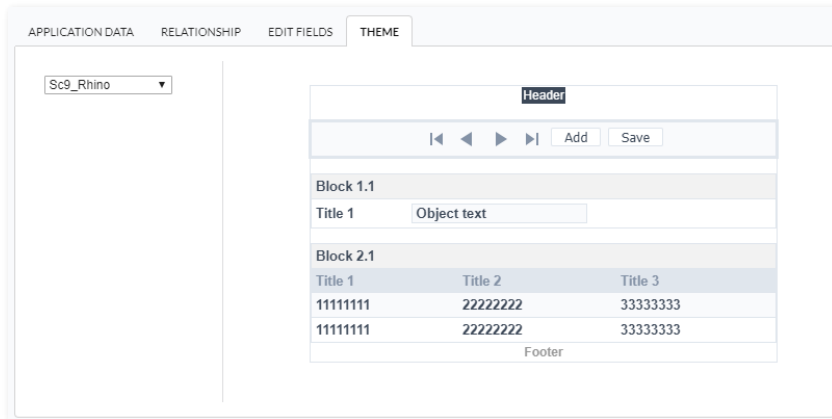
You just need to select which application you want to generate in the applications available in Scriptcase, That it will be generated automatically.



- **ame** - Application's name that is being created, It can not contain special characters.
- **ocalization** - Application's Language. The project's default language will be automatically selected.

Theme

In this screen you select the theme that the application will use. The default project's theme, that can be checked and changed in **Project > Properties**, will be selected automatically.

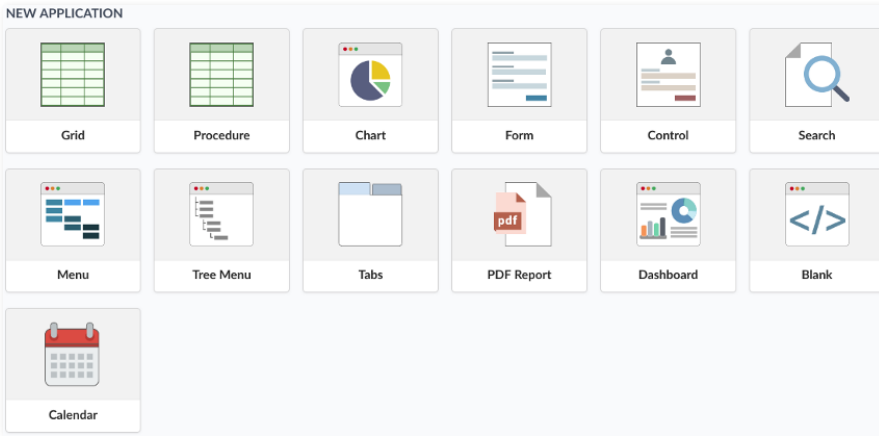


- **heme** - List of available themes in the project.

*After all the settings are done, click in the button **Create**. In the next screen the settings of the created application will be displayed.*

The creation process of one application is simple enough and intuitive.

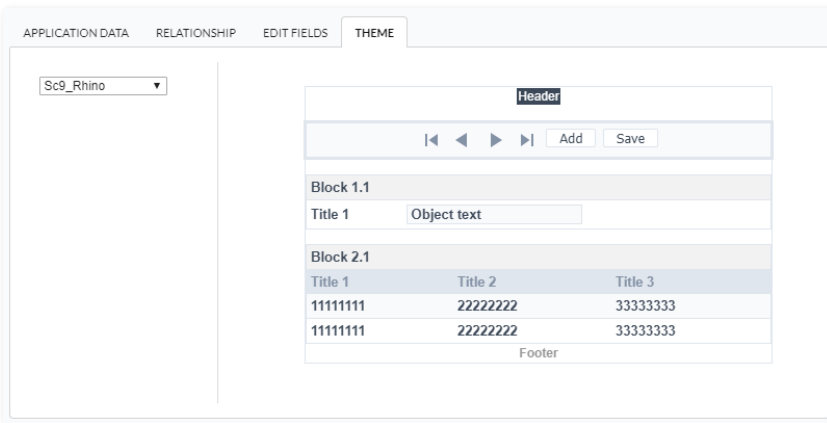
You just need to select which application you want to generate in the applications available in Scriptcase, That it will be generated automatically.



- **ame** - Application's name that is being created, It can not contain special characters.
- **ocalization** - Application's Language. The project's default language will be automatically selected.

Theme

In this screen you select the theme that the application will use. The default project's theme, that can be checked and changed in **Project > Properties**, will be selected automatically.



- **heme** - List of available themes in the project.

After all the settings are done, click in the button **Create**. In the next screen the settings of the created application will be displayed.

Through these attributes you can define some general search settings such as alignment, width, search conditions, etc.

ATTRIBUTE	VALUE	DESCRIPTION
Search Criteria	AND	Condition for searches with more than one field.
Display Condition	<input type="checkbox"/>	Display the search criteria.
Horizontal Alignment	Center	Defines the horizontal alignment of the application.
Friendly URL	<input type="text"/>	Friendly URL
Margins	<input type="text"/> Up <input type="text"/> Down <input type="text"/> Right <input type="text"/> Left	Set the margin.
Keep Values	<input checked="" type="checkbox"/>	Maintain the last search values when accessing the search page again.
Keep columns and sorting	<input checked="" type="checkbox"/>	Keeps last columns and order selection
Table Width	<input type="text" value="0"/>	Width value for the application table.
Table Width Unit	Automatic	Measure unit for the width.
Use Iframe	<input type="checkbox"/>	Use iframes to display the Search and the Grid on the same page.
Show Results	<input type="checkbox"/>	Display the search results on the same page when loading the application for the first time using iframes.
Iframe Height	<input type="text" value="1500"/>	Iframe height in pixels where the grid will be displayed.
Case Sensitive	<input checked="" type="checkbox"/>	Use case sensitive.
Use auto-complete in the fields.	Defined in the field	Field turns auto-complete automatically based on the existing values in the database.

Search settings interface.

- **Attributes**

- **Search Criteria** : Using this option you can select the SQL command as "AND" or "OR" to set the search criteria to be used;
- **Display Condition** : If you choose "Yes", it will be displayed all records when the search fields are blank. If the option set is "No", no record will be displayed;
- **Horizontal Alignment** : This option sets the application horizontal alignment;
- **Friendly URL** : This field allows you to change the URL that will be called by the application. Allowed characters are the same available on URLs: a-z, A-Z, 0-9, -_. This option can also be changed on the home screen, at the "Friendly URL" column at the applications list;
- **Margins** : Allows you to set the Search application positioning (margins).
- **Keep Values** : When this option is activated the last searched values are going to be maintained and displayed when the system user returns to the search application screen.
- **Keep columns and sorting** : This option determines if each search should preserve the selected values to the Grid fields columns and sorting, i.e. to each search, these values return to original condition.
- **Table Width** : This option sets the search application HTML width value, it can be in pixels or percent. You need to set the unit within the option "Table Width Unit".
- **Table Width Unit** : Using this option you can set the Search application width unit;
- **Use Iframe** : When this option is enabled the Search screen and the search results are going to be displayed in the same browser window, using two FRAMES, one above the other. The Search application and the Grid with the results are going to be part of the same page.
- **Show Results** : This option is associated with the "Use Iframe". When enabled it will initially displays the Search application along with the Grid inside the iframe positioned below the search screen, when disabled initially displays only the Search screen.
- **Iframe Height** : This option is also associated with the "Use Iframe". It sets the height, in pixels, of the iframe where the recovered data will be displayed by the search.
- **Case Sensitive** : This option enables the Case Sensitive option.
- **Use auto-complete in the fields** : Using this option you can enable the auto-complete option automatically based on the existing values in the database tables. You can also define this option one by one using the "defined in the field"

Select fields

Select fields interface.

Required

This interface has the options for you to select and set the Search application required fields.

Required fields interface.

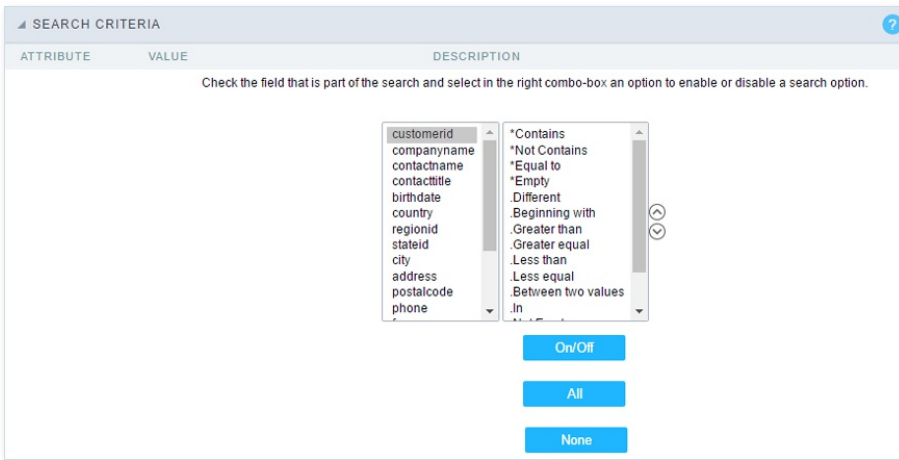
Within the generated application will be displayed a marker(*) next to the field and an error message will also pop up if there's no value assigned to the mandatory fields. You will be able to set some options for the market using the options below.

Marker placement configuration interface.

- **arker position** : Marker's position relative to the field (options are right, left or Do not display).
- **isplay message** : Displays whether or not the validation error message.

Search Criteria

With this interface, you can configure the conditions available for each field of the Search form.



Search configuration Interface.

We can see the fields list on the left combo. On the right, the list of options for filtering the selected field. To select an option, click on one of them (Equal to, Beginning with, Contains, etc.) and then the button On/Off. The arrows, on the right, allows altering the order of the fields.

For the Date type fields, you can define special conditions for the search, accessing the field configurations, and editing the Special Conditions Settings.

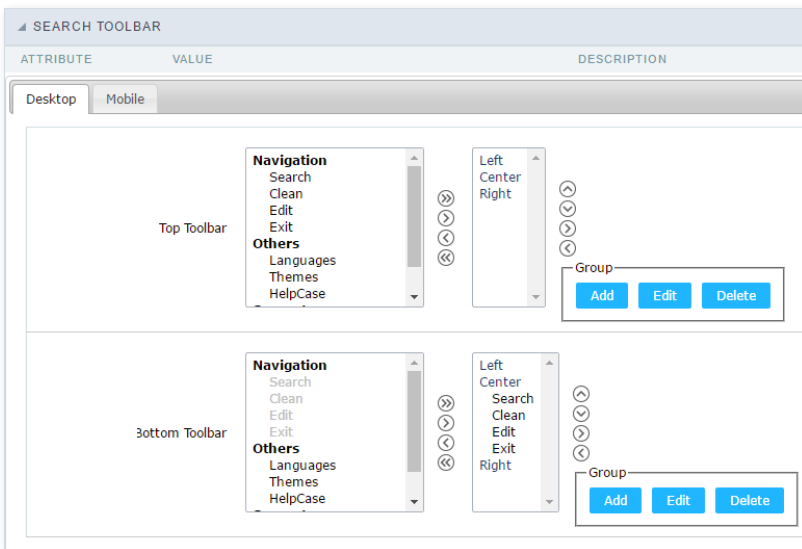
Below the list are the buttons to enable the selected options:

- **n/Off**: Enables or disables the field or the option chosen.
- **ll**: Marks all fields or options.
- **one**: Unmarks all the fields or options.

Toolbar

Desktop

The Search toolbar is divided in two parts: Top and Bottom, in a way that is possible to define the buttons that will be displayed in both bars. The selection of buttons in the top and bottom toolbar works independently, allowing the buttons to be displayed in both bars at the same time.



Toolbar Interface.

Navigation: Groups the options relative to the navigation buttons that can be displayed in the application.

- **earch**: Execute the search.
- **lean**: Clean the all the search fields.
- **dit**: Enable the *Save Tag* option.
- **xit**: Exit the application.

Others: Groups a diversity of options relative to the application.

- **anguages**: Displays a combobox with the names available, defined in the project properties.
- **hemes**: Displays a combobox with the themes available, defined in the project properties.
- **elpCase**: Displays a button to redirect to the help page.

Separator

- : Displays a line separating the buttons, when used the Group Buttons.

Use in-line buttons: Allows the alignment of the filter buttons next to the fields.

- **_ Inline buttons_**: Allows you to select which buttons will be displayed next to the field, and you can sort them according to your wishes. This option is available by enabling **Use Online**

Buttons in the button settings.

Button Settings

Button	Label	Hint	Shortcut key
Search			
Clean			
Edit			
Exit			

Button Settings Interface.

- **otkey**: Allows you to set keyboard hotkeys to a button.
- **se Inline Buttons**: Allows alignment of the filter buttons next to the fields.
- **osition of the in-line buttons**: Sets the positioning of the buttons to the right or left of the fields.
- **olumn Quantity**: Sets the number of display columns of the buttons, allowing you to configure whether they will be displayed side-by-side or distributed in columns.

Options

ATTRIBUTE	VALUE
Buttons Position (top)	Center ▾
Buttons Position (bottom)	Center ▾

Options Interface.

- **utton Position(Top/Bottom)** Positioning the buttons of the toolbar Top/Bottom.

Save Search

This feature allows the end-user to save his searches in a profile. You can create some rules, like to save the searches by user login.

SAVE CRITERIA	
<input checked="" type="checkbox"/> Enable Public	
Title (lang_srch_public)	<input type="text"/>
<input checked="" type="checkbox"/> Use Rules	
	<input type="button" value="New rule"/> <input type="button" value="Edit"/> <input type="button" value="Delete"/>

Save Filter Interface.

SAVE CRITERIA	
Name	<input type="text" value="var_login"/>
Variable name	<input type="text"/>
Label	<input type="text"/>
	<input type="button" value="Add"/> <input type="button" value="Delete"/>
	<input type="button" value="Finish"/> <input type="button" value="Cancel"/>

Save Filter Interface.

Save Tag

This feature allows the end-user to save his tags of searches in a profile. To save a search, he must click on the "Edit" button, then add a name to the profile and click on "Save Filter".

<input type="button" value="Orderid: Greater than 10,250 x"/> <input type="button" value="+ Add filter"/> <input type="button" value="Edit"/> <input type="button" value="x"/>
--

Search Tags.

<input type="button" value="Save filter"/> <input type="button" value="Cancel"/>
<input type="text"/> <input type="button" value="Save"/>

Save Tag Interface.

Layout

Blocks

Blocks are “containers” where you can position the application fields of Forms, Controls, or Slide Grids.

Scriptcase creates applications with one block by default. You can add more blocks as you wish, to organize it in the best way.

See below, the Columns Organization, and where you can define the position of the next block: beside or below the current one.

Block		Title		Label		Fields		Organization		
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
	form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Pag2										
	Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open

Blocks not shown

Application Block configuration

On the left side of each block, there are two icons, the first one to edit the information of the block and the second one to delete the block.

Organizing The Position Of The Blocks

See below how to modify the display order of the Blocks in one Page.

Click and drag the block that you desire to modify to its new position.

Block		Title		Label		Fields		Organization		
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
	form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Pag2										
	Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open

Blocks not shown

Application Block Display configuration

- see how to remove a block from the display

Click on the block desired and drag it to the item “Blocks not Shown”. This way, you can also drag the block to another page if desired. See the images below.

Block		Title		Label		Fields		Organization		
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
	form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Pag2										
	Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Blocks not shown										
	Block 3	Block 3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open

Application Block Display configuration

Block		Title		Label		Fields		Organization		
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
	form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Pag2										
	Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Blocks not shown										
	Block 3	Block 3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open

Application Block Display configuration

Block

- **ame:** The name that identifier the Block.
- **abel:** Title of the block to display in the application.

Title

- **isplay:** It controls the display of the block title.

Label

- **isplay:** It controls the display of the field labels of the block.
- **osition:** Options to display label :
 1. **bove:** Display the label above the field.
 2. **eside:** Display the label beside the field.
 3. **elow:** Display the label below the field.

Fields

- **olumns:** Amount of columns side by side in the block.
- **osition:** The way to display the fields in the block:
 1. **elow:** Display the fields one below the other respecting the number of columns.
 2. **eside:** Display the fields one beside the other respecting the number of columns.
 3. **ine:** Display the fields one beside the other with no tabulation.

Organization

- **ext**: The way to display the blocks in the page:
 1. **elow**: Set to show the following block below the current one.
 2. **eside**: Set to show the following block beside the current one.
 3. **abs**: Set to show the following block in a different tab then the current one.
- **idth**: Set the block width in pixels or percentages. Use the symbol "%" to indicates the value in percentage.
- **ollapse**: Enables the option to close the block.

Create A New Block

To include new blocks in an Application, click on the button [Create New Block](#). Then, enter the name and label of the block in the following interface and finish by click on Create.

Creating application blocks configuration


Name

Name of the Block.

Label

Title of the block to display in the application.

Edit Blocks

To edit a block, click on the icon , that is on the left side of the block. Then you can see the following interface to define the parameters of the blocks. Click on Save to finish.

ATTRIBUTE	VALUE
Name	<input type="text" value="form_orders"/>
Title	<input type="text" value="form_orders"/>
Display Title	<input type="radio"/> Yes <input checked="" type="radio"/> No
Title Font	<input type="text"/> Aa
Font Size	<input type="text"/>
Font Color	<input type="text"/>
Background Color	<input type="text"/>
Background image	<input type="text"/>
Title Height	<input type="text" value="20"/> pixels
Horizontal Alignment	<input type="text"/>
Vertical Alignment	<input type="text"/>
Display Label	<input checked="" type="radio"/> Yes <input type="radio"/> No
Columns	<input type="text" value="1"/>
Columns Width	<input type="text" value="Calculated"/>
Label Color	<input type="text"/>
Fields Organization	<input type="text" value="Beside"/>
Label Position	<input type="text" value="Beside"/>
Next Block	<input type="text" value="Below"/>
Border Color	<input type="text"/>
Border Width	<input type="text" value="0"/> pixels
Block Width	<input type="text" value="100%"/>
Block Height	<input type="text"/>
Cell Spacing	<input type="text"/> pixels
Collapse	<input type="text" value="Start open"/>

Application Block editing interface

Name

Name of the block. ##### Title
 Block title for display. ##### Display Title
 This option, when active, allows displaying the block title. ##### Title Font
 Set the font family of the block title. ##### Font Size
 Set the font size of the block title. ##### Font Color
 Set the font color of the block title. ##### Background Color
 Set the Background Color of the block title. ##### Background image
 Set a Background image for the block title. ##### Title Height
 Height in pixels of the block title line. ##### Horizontal Alignment
 Horizontal Alignment of the block title (Left, Center, and Right). ##### Vertical Alignment
 Vertical Alignment of the block title (Top, Middle, and Bottom). ##### Display Label
 Display the labels of the fields in the block. ##### Columns
 Amount of field columns in a block. ##### Columns Width
 Set the field column width of the block. ##### Label Color
 Color of the field labels. ##### Fields Organization
 The way to display the fields in the block. ##### Label Position
 Set the position of the field labels of the block. ##### Next Block
 Set the position of the following block relating to the current one. ##### Border Color
 The border Color for the block. ##### Border Width
 The border Width for the block. ##### Block Width
 The width for the block. ##### Block Height
 The Height for the block. ##### Cell Spacing
 The Cell Spacing in the block. ##### Collapse
 It enables the option to close the block.

This module lets you editing display options for the application. You can apply visualization themes, organize blocks, define values, and the display format of the Header and Footer.

Layout Settings

ScriptCase creates an app by using default values defined per project, but you can change those values for a specific application on this interface.

▾ LAYOUT SETTINGS

Header Template	<input type="text" value="Flat"/>	Template name used for the application header.
Footer Template	<input type="text" value="Default"/>	Template name used for the application footer.
Button	<input type="text"/>	Use different buttons than what was defined in the color scheme.
Themes	<input type="text" value="Sc9_Rhino"/>	Use different themes from the one defined for the color scheme

Header

xxxyzz xxxxx yyyy ▾

Block 1

Name

Type Male Female

Address*

Groups* Male Female

Countries ▾

Address

Drag & Drop files here

Photos

Image1.png ✓
Image2.png ✗

Captcha

Application Layout Settings

Header Template

Allows choosing the look of the application Header.

Footer Template

Allows choosing the look of the application Footer.

Button

Allows choosing the button theme for the application.

Themes

Choose one of the selected themes in the project properties. It defines the look of the application, like colors, fonts, and others.

Header & Footer

Header

Here we can define if we want to display the header or not, the application title, and fill the header variables. The header looks according to the selected template in the Layout settings interface.

HEADER	
ATTRIBUTE	VALUE
Display Header	<input checked="" type="checkbox"/>
Insertion Title	{lang_othr_frmi_titl} - orders
Update Title	{lang_othr_frmu_titl} - orders
Header Variables	
SC_HEADER	Title
SC_VALUE	Date

Application Header Configuration

Display Header


This option determines if we want to display the header or not.

Title

Allows informing the title of the application.

Header Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **itle**: It displays the value of "**Application Title**" in the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Depending on the Application, you may have more than one title option.

Footer

FOOTER	
ATTRIBUTE	VALUE
Display Footer	<input type="checkbox"/>
Footer Variables	
NM_FOOTER_DATA	
NM_FOOTER_LOGO	
NM_FOOTER_TITLE	

Application Footer Settings


This page may change depending on the footer template chosen in the Layout Settings.

Display Footer

This option determines if we want to display the footer or not.

Footer Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

SQL Settings

This interface allows configuring the related database settings, such as the SQL statement, the used database connection, case sensitive, and others.

ATTRIBUTE	VALUE	DESCRIPTION
SQL Select Statement	<pre>SELECT customerid, companyname, contactname, contacttitle, birthdate, country, regionid, stateid, city,</pre>	
Limit	<input type="text"/>	It sets the number of records to be retrieved from the SQL statement.
SQL Preparation	<input type="text"/>	
Connection	<input type="text" value="conn_example"/>	Connection name to access the database.
Use Customized Message	<input type="checkbox"/>	Use a customized error message when the application has no records.
No Records Message	<input type="text"/>	When the application has no records, it will display this customized text.
Font	<input type="text" value="Aa"/>	Font face of the error message.
Font Size	<input type="text" value="12"/>	Font size of the error message.
Font Color	<input type="text" value="#000000"/>	Font color of the error message.
Variable for Table	<input type="text"/>	Variable name used for replacing the table name. Please indicate the name of the table that will be replaced by the variable value.
Fields Variables	<div style="border: 1px solid black; padding: 2px;"> <p>Variable</p> <input type="text"/> <p>customerid</p> </div>	Variables for substitution of the field names on the application. For each dynamically determined field, inform the name of the variable and the field that will be substituted.
Case Sensitive	<input checked="" type="checkbox"/>	Use case sensitive.

Grid SQL configuration

SQL Select Statement

It allows you to define the primary SQL of the application. You can edit this SQL to add or delete fields.

Limit

Lets you limit the display in the number of records retrieved by SQL query.

SQL Preparation

You can enter SQL commands or procedure names to execute them before the primary SQL of the application.

Connection

It allows defining the database connection of the application. You can change the connection to another one that has the same table.

Use Customized Message

Lets you define to display the "no records" message or not.

No Records Message

Lets you set the message when the application has no records.

Font

This option is available when using the "Use Customized Message" option. It lets you set the font for the message.

Font Size

This option is available when using the "Use Customized Message" option. It lets you to set the font size.

Font Color

This option is available when using the "Use Customized Message" option. It lets you to set the font color.

Variable for Table

It allows to use a variable to change a part of the string containing the table name.

Fill the first input with the name of the variable (replace to). The second input you should fill with the part of the name of the table you want to replace (replace from).

Fields Variables

It allows to use a variable to change a part of the string containing the field name.

Fill the first input with the name of the variable (replace to). The second input you should fill with the part of the name of the field you want to replace (replace from).

Case sensitive

It defines if the database connection uses case sensitive or not.

Security ?		
ATTRIBUTE	VALUE	DESCRIPTION
Use Security	<input type="checkbox"/>	Application needs validation to run. Read about the sc_apl_status macro.
Url output of the security	<input type="text"/>	Uri to be redirected if the user does not have access to this application.
Use Password	<input type="checkbox"/>	Enter password required to run the application.
Request password just once	<input type="checkbox"/>	Request password just once per session without asking for password every time you open the application in the same session.
Enable direct call by URL	<input checked="" type="checkbox"/>	Allow an application to be called by typing the URL directly in the browsers.

Application Security Configuration

Use Security

Turns the application accessible just when using the macro sc_apl_status or using the security module. When it is enabled, the application displays a message "User not Authorized" if the user tries to access it directly.

Url output of the security

It redirects the application to another URL after displays the warning "User not Authorized".

Use Password

Turns the application accessible just by using a password.

Request password just once

Request the password once per session, not needing to inform the password every time you access the application.

Enable direct call by URL

Allows to call the application directly from the URL in the browsers.

Allows calling the application directly from the URL in the browsers. In the development environment, you can disable those options for testing, it avoids requiring the login every time you run the application. To disable the security and password in development, you must access Options > My ScriptCase and uncheck the options: Enable Use of Security, Enable Use of Password.

This interface allows you to define a Log schema to the app. The Log schema tracks what the end-users are doing in the app. Those are the events it can tracks: Access, Insert, Update, and delete.

To learn how to create a Log schema [see here](#).

The screenshot shows a configuration window titled "LOG". It features a table with two columns: "ATTRIBUTE" and "VALUE". The "Schema" attribute has a dropdown menu currently set to "log". Below the table, there are two empty list boxes: "Events" on the left and "Access" on the right. Between these two boxes are two blue buttons: "Add >>" and "Remove <<".

Application Log configuration

Schema Select an existing log schema for the application.

Events You can select just specific events to save in the log.

1. [Home](#)
2. Applications

General Overview

The applications can have two types of fields.

Table Fields

Columns of the table connected to the application and their data types.

Example:

In a table with the following structure:

```
CREATE TABLE `orders` (
  `orderid` int(11) NOT NULL,
  `customerid` varchar(5) NULL DEFAULT NULL,
  `employeeid` int(11) NULL DEFAULT NULL,
  `orderdate` date NULL DEFAULT NULL,
  `requireddate` date NULL DEFAULT NULL,
  PRIMARY KEY (`orderid`) USING BTREE
);
```

These are the listed fields on left menu **Fields**:

Virtual Fields

Fields created inside application to assist in the development process.

How to create a virtual field

1. Inside an application, access the left menu **Fields** and click on **New Field** option.
2. Choose how many fields do you want to create:

1. Create the fields and define type, name and label for each one:

- **Type:** Data type of the field.
- **Name:** Internal name of the field. Used to identify the field on events and Scriptcase interface.
- **Label:** Title of the field displayed on the application.

Scriptcase **doesn't insert** virtual fields on the table after its creation.

Types of fields by applications

Application Table Fields Virtual Fields

Grid	<input type="text"/>	<input type="text"/>
Procedure	<input type="text"/>	<input type="text"/>
Chart	<input type="text"/>	<input type="text"/>
Form	<input type="text"/>	<input type="text"/>
Control	<input type="text"/>	<input type="text"/>
Search	<input type="text"/>	<input type="text"/>
PDF Report	<input type="text"/>	<input type="text"/>
Calendar	<input type="text"/>	<input type="text"/>

The applications **Menu**, **Tree Menu**, **Dashboard** and **Blank** were not listed because they haven't manage fields.

□

1. [Home](#)
2. [Applications](#)
3. Search
4. Text

Text

General Settings

This type of field allows the developer to create quickly inputs to insert and update data, where the final user can inform its data to be allocated in its database.

Type of Data

Defines the type of the field in the application. In this case we should select **Text**.

Search Label

Defines the title that will be displayed in the field when the application is executed. The terminology used in the interface is fundamental to the system have a good usability, who should use names and familiar terms to the final user of the application, instead of using terms from the system.

For example, this text field that has the name `sellername_`, **the client would have a much better understanding of the functionality of the field when we define the label as `_Seller name`.**

Not only a fixed text, the **Label** attribute allows the use of langs to define the field's title, making it possible to use your application in a multi language project.

Watermark

Informing a text to the **Watermark** it will be displayed in the input a text as an example that can be informed in the field. The result after set will be this:

Amount of Characters

Allows to set the width of the text field's input that varies with the amount of characters informed. Although, if the amount of characters typed are greater than the setting, the text will be pushed to the left, to keep the maximum amount of characters as defined.

Maximum Size:

Allows the user to determinate the maximum size used in the application field.

SQL Type

Informs the type of the field in the database.

Field Behavior

Use autocomplete:

Field automatically turns into autocomplete according to existing values in the database.

Use Select2

Uses the new component for data selection, allowing searches within the select comboBox.

Width for the Select2

Sets a width for the area in the Select2.

Amount of characters

Sets the amount of characters to start the search.

Amount of rows

Sets the maximum number of rows to list the search result.

Width:

Defines a width in pixels for a result box.

Search options:

Defines the validation that will be made to fetch the search result.

Start equals to: Will return the records with the same start value as in the database.

Any part: Will return the records when exist the character in any part of the record.

End equals to: Will return the records with the same final value as in the database.

Position between values:

Defines the position that objects will be displayed.

Text Between Values

Text that will appear when using a filter condition between two values.

OnChange Submit:

Submit search on this field changing.

Show Condition:

To show or not the search condition. It only works if the search has at least one condition.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

1. [Home](#)
2. [Applications](#)
3. Search
4. Integer

Integer

On this page, you can configure settings related to your field of type Number. Since the use of specific symbols display until the mode in which they are displayed. And thus, boost your application.

Data type:

It sets the application field type. When the field is set as Number, it is permissible to define formatting rules of integers.

Watermark

Informing a text for the **watermark** it will be displayed on the input text with an example of what can be reported in the field. The result after that will be configured:

Use slider:

It displays a slider component in the field. So you can increase or decrease the value sliding the cursor. You can also customize the increment value, if it increments the value 1 by 1, 2, 5, 10... N.

Amount of characters

It allows you to set the width of the input text field according to the amount of characters. However, if the quantity entered is greater than the set for the characters, the text will be pushed to the left, in order to ensure the maximum amount of characters set in the option of **Values formatting**.

Tipo SQL:

Reports the SQL type of the field as it is configured in the database.

Use autocomplete:

The field turns into autocomplete automatically according to the values in the database.

Amount of characters:

Sets the amount of characters to start the search.

Number of rows:

It sets the maximum number of rows to list the search result.

Width:

it sets the width in pixels for the results.

Search options:

It sets the validation that will be made to get the search result.

Position between values:

It sets the position that objects will be displayed.

Text between values:

Text that will appear between the values.

Submit in Onchange:

Allows you to submit the filter when there are changes in the field.

Display Condition:

Allows you to display or not the condition of the filter, it will only work if the filter has at least an option.

Use regional settings:

It allows you to apply regional settings to the number fields formatting. When this option is not selected it will be displayed the attributes group, minus sign and negative number format.

Grouping:

It allows you to define which character is used to separate thousands. This option is only available when the option to use regional settings is disabled.

Accept:

It allows to define if the field accept only negative values, positive-only or both.

Lookup Settings

The Lookup is a way used to provide the end user a list of values description/name where the items can be selected and used in the application. For example, for a field sex that in

the database is represented with M or F values can be presented as male or female dynamically with the lookup. So you will be able to retrieve these values from a database table or manually, if the reference table does not exist.

Example:

Lookup Methods

There are two lookup methods available:

Manual: To use this method it is necessary to inform the selection conditions manually to the Lookup.

Automatic: To use this method it is necessary to use a lookup query from the database to access the reference table.

Editing-automatic lookup:

Select Command:

It sets the SQL command that will retrieve the value to display inside the Grid field. The command must have the following format

```
SELECT field_to_be_displayed FROM table WHERE key_field = {grid_field}
```

The Grid field must be always referenced between curly braces {}, So at run time, the field braces will be replaced by its value in the table.

Multiple Options:

When the Select command informed return multiple values, the option must be selected.

Delimiter:

It sets the tab of values returned by the Select command, this option should be filled when the "Yes" option is checked in the multiple attribute options.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

Use a specific connection:

This option allows you to select another existing connection in the project, other than the current connection, to select the values for the lookup.

Edit Lookup - Manual:

Lookup used to modify the display of the field with predefined values.

Lookup Types

It sets the operating mode of the lookup can be: single value, multiple values (delimiter), multiple values (position) and multiple values (binary).

Single value: it will be displayed only one value for the field. There is a direct replacement of the value stored in the Bank for a label defined in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value of "M" will be replaced by "Male".

Multiple Values (Delimiter):

By using this option it will be displayed multiple values for the selected field. The values must be separated by a delimiter to be informed. A replacement of parts from a string stored in the field, separated by a delimiter for values contained in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value S;M will be replaced by Sports and Movies.

Delimiter: Character used to separate the values within the same string.

Multiple Values (Position)

Allows you to recover information stored in a single string of the selected field. In order for this information to be retrieved must be informed, in addition to the label, the starting

position and the number of bytes that each information occupies within the string.

As an example we will use a string to store Sex, Marital status and Hobby respectively. Sex occupies one byte, Status one byte and Hobby two bytes .

For this purpose we define the list as:

Label	Value	Start	Size
Male	M	1	1
Female	F	1	1
Married	M	2	1
Single	S	2	1
Study	ST	3	2
Sportes	SP	3	2
Reading	LE	3	2

Example: the string MMST would be presented in the query as: Male Married Study

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value M will be replaced by Male.

Start: : Starting position in the string where the information is recorded. The first position is always 1.

Size : Number of bytes that the information occupies in the string.

Multiple values (binary):

It allows you to retrieve several information stored in decimal form of the selected field.

As examples we will use the following list (although not informed in the interface of inclusion of values in the list, each item has a value assigned automatically according to the order):

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

To display the data, a decomposition of the decimal number stored in the bank is performed. For example, numbers 11 and 12 (written in the database) would be decomposed as follows:

$$11 = 1 + 2 + 8 = (\text{Sports} - \text{Culture} - \text{Reading})$$

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

$$12 = 4 + 8 = (\text{Leisure} - \text{Reading})$$

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

Label:

Text that will be displayed in the Grid.

Include Button:

Includes in the list the values populated in the Label and value fields.

Alter Button:

It changes the attributes of the selected item.

Delete Button:

Deletes the selected item.

Clean Button:

It cleans the fields

Save Button:

Allows the user to save the entire list, for later use using the Load Lookup definition option.

Load Lookup definition:

Allows you to load a predefined value list for use as a lookup.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**

- **Font** : Allows to choose the font type, that will be applied to the application field title.
- **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
- **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.

- **CSS of the Field**

- **Font** : Allows to choose the font type, that will be applied to the application field.
- **Font Size** : Allows to choose the the font size, that will be applied to the application field.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
- **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the field.
- **Height** : To define a height for the field.

- **CSS of the Input Object**

- **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
- **Font Size** : Allows to choose the the font size, that will be applied to the application field.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the Input Object.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
- **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the Input Object.
- **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Search
- 4. Decimal

Decimal

General Settings

Decimal field Configuration Interface.

- **Data Type** : Select the type of field for the application. When it is defined as a Decimal you can define the format of a decimal number.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Accumulated** : When enabled, define that the field will be accumulated with the values of another field. This option is only available on virtual fields.
- **Accumulator field** : Only available when the option "Accumulated" is enabled. This configuration defines which field will be used to accumulate the values. The chosen fields needs to be Integer, Currency or Decimal.
- **Grid Mask** : Defines the mask for the field display. There are two typed of masks described below:

Character	Description
X	Placeholder to any character. Replaced by any character. If number of characters entered are less then the mask size, the field value is completed with zeros (Filling full size field entry is required).
Z	Replaced by any character retrieved from database. Suppress zeros at field left (Complete field filling is optional). When used combined with the mask character X it should be placed at the mask left.

Mask Examples

Mask	Field Value	Formatted Value
(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
(zz) xxxx - xxxx	1234567890	(12) 3456 - 7890
(z) xxxx - xxxx	12345678	() 1234 - 5678
(zz) xxxx - xxxx	0012345678	() 1234 - 5678

- **Do Not Repeat Value** : Does not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Field Behavior

Use autocomplete:

Field automatically turns into autocomplete according to existing values in the database.

Use Select2

Uses the new component for data selection, allowing searches within the select comboBox.

Width for the Select2

Sets a width for the area in the Select2.

Amount of characters

Sets the amount of characters to start the search.

Amount of rows

Sets the maximum number of rows to list the search result.

Width:

Defines a width in pixels for a result box.

Search options:

Defines the validation that will be made to fetch the search result.

Start equals to: Will return the records with the same start value as in the database.

Any part: Will return the records when exist the character in any part of the record.

End equals to: Will return the records with the same final value as in the database.

Position between values:

Defines the position that objects will be displayed.

Text Between Values

Text that will appear when using a filter condition between two values.

OnChange Submit:

Submit search on this field changing.

Show Condition:

To show or not the search condition. It only works if the search has at least one condition.

Values format

Decimal Field Format of Values with Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the number of the field. When not enabled, the attributes Digit Grouping Separator, Decimal Separator, Negative sign and Negative number format.
- **Decimal Precision** : Defines the amount of decimal places for the field.
- **Digit grouping** : Defines if the field will display the digits separator.
- **Decimal Separator** : Defines the decimal separator.
- **Accept** : Determines if the field, will accept only negative, positive or both numbers.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

1. [Home](#)
2. [Applications](#)
3. Search
4. Currency

Currency

General Settings

![[Currency field Configuration Interface.][filtro_cons_moeda] *Currency field Configuration Interface.*

- **Data Type** : Define the type of field for the application. When set to Currency, you can currency values to the field.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Field Behavior

![[Currency Field Behavior Interface of the Search Configuration.][filtro_cons_moeda_filtro] *Currency Field Behavior Interface of the Search Configuration.*

- **Use auto-complete** : The field behaves as an auto-complete according to the values existing in the database.
- **Amount of characters** : Sets the amount of characters to start the search.
- **Amount of rows** : Sets the maximum number of rows to list the search result.
- **Width** : Sets the width in pixels for the result box.
- **Search options** : Defines the validation that will be made to fetch the search result.
- **Position between values** : This option sets the position that objects will be displayed.
- **Text between values** : Text that will appear between values.
- **OnChange Submit** : When enabled, it submits the search when there is modifications to the field.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Values format

![[Currency Field Format of Values with Regional Settings.][filtro_cons_moeda_format] *Currency Field Format of Values with Regional Settings.*

- **Regional Settings** : Allows to apply the Regional Settings to format the number of the field. When not enabled, the attributes Digit Grouping Separator, Decimal Separator.
- **Decimal Precision** : Defines the amount of decimal places for the field.
- **Digit grouping** : Defines if the field will display the digits separator.
- **Decimal Separator** : Defines the decimal separator.
- **Accept** : Determines if the field, will accept only negative, positive or both numbers.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Search
- 4. Date

Date

General Settings

- **Data Type** : Define the type of field for the application. When set to Date, you can inform a date.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Watermark**: Displays a watermark in the field input.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the datatype of field in the database.

Values format

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator attribute.
- **Date separator** : Allows you to inform the separator symbol for the date.
- **Display** : Select the format of the day for display.
- **Internal Format** : Allows to define the format the field when the SQL type is different from **Date**. You need to use the characters **A**, **M** and **D** that correspond to **Year**, **Month** and **Day**.
- **Use Combo-box** : Allows you to select the date using a combo-box.
 - **Year as Combo** : Allows to use the year combo to select the date.
 - **Initial Year** : First year displayed in the combo.
 - **Actual Year +** : Display the current plus the amount of years informed.
- **Month in full textual** : Displays the Month format in Full.
- **Display Calendar** : Enables the a calendar icon beside the field, this allows to select the date from a calendar with the format already setup.
 - **New Calendar** : Defines if the JQuery calendar (New Calendar) is going to be displayed or the old format.
 - **Years Limit** : Amount of years displayed in the calendar.
 - **View week number**: Displays the number of the week in the application.
 - **Additional months** : Displays the additional months of the calendar.
 - **Show Combo year and month** : Displays the year and month of the calendar in the combo box.

Field Behavior

- **Position between values** : This option sets the position that objects will be displayed.
- **Text between values** : Text that will appear between values.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Special Conditions

- **All Period** : Searches for all periods of dates.
- **Today** : Searches in today's date.
- **Yesterday** : Searches in yesterday's date.
- **Last 7 days** : Searches the last 7 days. Ex: ((01/01/2017 01/07/2017)).
- **This month** : Searches the dates from the first day of the current month.
- **Last month** : Searches the dates from the first day of lasts month.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).

- **Width** : To define a width for the field.
- **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Search
4. Time

Time

General Settings

![[Time field Configuration Interface.]][[filtro_cons_hora] *Time field Configuration Interface.*

- **Data Type** : Define the type of field for the application. When set to Time, you can inform a time to this field.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Values format

![[Time Field Format of Values with Regional Settings.]][[filtro_cons_hora_filtro] *Time Field Format of Values with Regional Settings.*

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator and time separator attributes.
- **Time separator** : Allows you to inform the separator symbol for the time.
- **Display** : Select the format of the time for display.
- **Internal Format** : Allows to define the format the field when the SQL type is different from **TIME**. You need to use the characters **HH**, **II**, and **SS** that correspond to **Day**, **Hour**, **Minutes** and **Seconds**.

Field Behavior

![[Time Field Behavior Interface of the Search Configuration.]][[filtro_cons_hora_format] *Time Field Behavior Interface of the Search Configuration.*

- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

1. [Home](#)
2. [Applications](#)
3. Search
4. Datetime

Datetime

General Settings

![[Datetime field Configuration Interface.]][[filtro_cons_data_hora] *Datetime field Configuration Interface.*

- **Data Type** : Define the type of field for the application. When set to Datetime, you can inform a date and time to this field.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Values format

![[Datetime Field Format of Values with Regional Settings.]][[filtro_cons_data_hora_format] *Datetime Field Format of Values with Regional Settings.*

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator and time separator attributes.
- **Date separator** : Allows you to inform the separator symbol for the date.
- **Time separator** : Allows you to inform the separator symbol for the time.
- **Display** : Select the format of the day/time for display.
- **Internal Format** : Allows to define the format the field when the SQL type is different from **DATETIME**. You need to use the characters **A, M, D, HH, II, and SS** that correspond to **Year, Month, Day, Hour, Minutes and Seconds**.
- **Use Combo-box** : Allows you to select the date using a combo-box.
 - **Year as Combo** : Allows to use the year combo to select the date.
 - **Initial Year** : First year displayed in the combo.
 - **Actual Year +** : Display the current plus the amount of years informed.
- **Month in full textual** : Displays the Month format in Full.
- **Display Calendar** : Enables the a calendar icon beside the field, this allows to select the date from a calendar with the format already setup.
 - **New Calendar** : Defines if the JQuery calendar (New Calendar) is going to be displayed or the old format.
 - **Years Limit** : Amount of years displayed in the calendar.
 - **View week number** : Displays the number of the week in the application.
 - **Additional months** : Displays the additional months of the calendar.
 - **Show Combo year and month** : Displays the year and month of the calendar in the combo box.

Field Behavior

![[Date Field Behavior Interface of the Search Configuration.]][[filtro_cons_data_hora_format] *Date Field Behavior Interface of the Search Configuration.*

- **Position between values** : This option sets the position that objects will be displayed.
- **Text between values** : Text that will appear between values.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Special Conditions

![[Datetime Field Special Conditions.]][[filtro_cons_data_cond] *Datetime Field Special Conditions.*

- **All Period** : Searches for all periods of dates.
- **Today** : Searches in todays date.
- **Yesterday** : Searches in yesterdays date.
- **Last 7 days** : Searches the last 7 days. Ex: ((01/01/2017 01/07/2017).
- **This month** : Searches the dates from the first day of the current month.
- **Last month** : Searches the dates from the first day of lasts month.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).

- **Width** : To define a width for the field.
- **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

1. [Home](#)
2. [Applications](#)
3. Search
4. Select

Select

General Settings

![[Select field Configuration Interface.]][[filtro_cons_select] *Select field Configuration Interface.*

- **Data Type** : Define the type of field for the application. When set to Select, you can select multiple option from a combo box (Select Field).
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Field Behavior

![[Select Field Behavior Interface of the Search Configuration.]][[filtro_cons_cpf_filtro] *Select Field Behavior Interface of the Search Configuration.*

- **Use autocomplete** : The field behaves as an autocomplete according to the values existing in the database.
- **OnChange Submit** : When enabled, it submits the search when there is modifications to the field.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Search Lookup

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

• Lookup Methods

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

Selecting the lookup type.

◦ Lookup Method - Automatic

Lookup used to list the values that will be displayed on the Checkbox field. These values will be recovered dynamically using the Select command.

Automatic Lookup Interface.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimitation.

- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter than (;).
- **Height** : Defines the height for the select object.
- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

- **Link** : Allows to create a link to another form allowing to manipulate the list displayed on the select field. After the manipulation, the select object it updated automatically.

◦ Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

Manual Lookup Interface.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).

- **Single Value** :

Setting up Single Value Lookup

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Default** : With this option enabled, the selected item will be inserted to the table when committed.
- **Height** : Defines the height for the select object.

■ Multiple Values (delimiter)

You can store various values for the select field. The values are separated by the informed delimiter. For example: the combination **Sport, Cinema and Tourism** selected in form will be stored like **E;C;T** in case the delimiter is ; (semi coma).

Setting up Multiple Values (delimiter)

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Height** : Defines the height for the select object.

■ Multiple Values (position)

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man, Single** and **Read**, in the database would be stored the following value **MSRD**.

Setting up Multiple Values (position)

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Negative** : Value to be saved in the table in case there is not any valued selected on the field.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.
- **Height** : Defines the height for the select object.

■ Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports, Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

Editing Lookup Configuration Multiple Values (Binary)

- **Label** : Text that will be displayed in the item list of the select field.
- **Height** : Defines the height for the select object.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.

- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

 Displaying the original lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Lookup Method - Actual value

This lookup is used to list all the values in the selected field.

This lookup will apply a “distinct” to your SQL query.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

1. [Home](#)
2. [Applications](#)
3. Search
4. Double Select

Double Select

General Settings

![[Double Select field Configuration Interface.]]filtro_cons_duplo_select] *Double Select field Configuration Interface.*

- **Data Type** : Define the type of field for the application. When set to Double Select, your allowed to have multiple options selected.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Field Behavior

![[Double Select Field Behavior Interface of the Search Configuration.]]filtro_cons_cpf_filtro] *Double Select Field Behavior Interface of the Search Configuration.*

- **Use autocomplete** : The field behaves as an autocomplete according to the values existing in the database.
- **OnChange Submit** : When enabled, it submits the search when there is modifications to the field.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Search Lookup

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or F can be presented like Male or Female or in a dynamic way, getting these values from the database.

Lookup Settings Display for the field. *Automatic Lookup Interface..*

- **SQL Select Statement** : Defines the SQL command that will get the values displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.
- **Height** : Set the height(lines) of the field interface.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter than (;).
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Selects another specific connection existing in the project. The select command will be done on the second connection.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.

- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the Input Object.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
- **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the Input Object.
- **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

1. [Home](#)
2. [Applications](#)
3. Search
4. Check box

Check box

General Settings

![[Check box field Configuration Interface.]][[filtro_cons_checkbox] *Check box field Configuration Interface.*

- **Data Type** : Define the type of field for the application. When set to Check box, your allowed to have multiple options selected.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Field Behavior

![[Check box Field Behavior Interface of the Search Configuration.]][[filtro_cons_cpf_filtro] *Check box Field Behavior Interface of the Search Configuration.*

- **Use autocomplete** : The field behaves as an autocomplete according to the values existing in the database.
- **OnChange Submit** : When enabled, it submits the search when there is modifications to the field.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Search Lookup

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or F can be presented like Male or Female or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).
Selecting the lookup type.

- **Lookup Method - Automatic**

Lookup used to list the values that will be displayed on the CheckBox field. These values will be recovered dynamically using the Select command.

Automatic Lookup Interface.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.

- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Option check all** : Displays two options on the field to check and uncheck all.

- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Selects another specific connection existing in the project. The select command will be done on the second connection.

- **Lookup Method - Manual**

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

Manual Lookup Interface.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).

- **Single Value** :

Setting up Single Value Lookup

- **Label** : Text that will be displayed in the item list of the checkBox.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Default** : With this option enabled, the selected item will be inserted to the table when committed.
- **Columns** : Set the amount of columns, for the list of items.

- **Multiple Values (delimiter)**

You can store various values for the checkBox field. The values are separated by the informed delimiter. For example: the combination **Sport, Cinema and Tourism** selected in form will be stored like **E;C;T** in case the delimiter is ; (semi coma).

Setting up Multiple Values (delimiter)

- **Label** : Text that will be displayed in the item list of the checkBox.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will

view in the list **Man**.

- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Columns** : Set amount of columns, for the list of items.

▪ **Multiple Values (position)**

Stores a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in the database would be stored the following value **MSRD**.

Setting up Multiple Values (position)

- **Label** : Text that will be displayed in the item list of the checkBox.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Negative** : Value to be saved in the table in case there is not any valued selected on the field.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.
- **Columns** : Set the amount of columns, for the list of items.

▪ **Multiple Values (binary)**

Stores a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute Value	Lookup Description
1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value	Lookup Description
1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value	Lookup Description
1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

Setting up Multiple Values (Binary)

- **Label** : Text that will be displayed in the item list of the checkBox.
- **Columns** : Allows you to inform the amount of columns, for the list of items.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Saves all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Refreshes the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Option check all** : Displays two options on the field to check and uncheck all.

- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Search
- 4. Radio

Radio

General Settings

![[Radio field Configuration Interface.]][[filtro_cons_radio]] *Radio field Configuration Interface.*

- **Data Type** : Define the type of field for the application. When set to Radio, your allowed to select one of the options listed.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Field Behavior

![[Radio Field Behavior Interface of the Search Configuration.]][[filtro_cons_cpf_filtro]] *Radio Field Behavior Interface of the Search Configuration.*

- **Use autocomplete** : The field behaves as an autocomplete according to the values existing in the database.
- **OnChange Submit** : When enabled, it submits the search when there is modifications to the field.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Search Lookup

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).
Selecting the lookup type.

- **Lookup Method - Automatic**

Lookup used to list the values that will be displayed on the Checkbox field. These values will be recovered dynamically using the Select command.

Automatic Lookup Interface.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.

- **Columns** : Allows you to inform the amount of columns, for the list of items.

- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
 - **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
 - **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

- **Lookup Method - Manual**

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

Manual Lookup Interface.

- **Label** : Text that will be displayed in the item list of the radio.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Default** : With this option enabled, the selected item will be inserted to the table when committed.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

Displaying the original lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

1. [Home](#)
2. [Applications](#)
3. Search
4. Text Auto-Complete

Text Auto-Complete

General Settings

![[Text Auto-Complete field Configuration Interface.][filtro_cons_texto_auto] *Text Auto-Complete field Configuration Interface.*

- **Data Type** : Define the type of field for the application. When set to Text auto complete, you can inform a value based on the select statement of the lookup settings and it will manipulate an internal Text for the data.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Field Behavior

![[Text Auto-Complete Field Behavior Interface of the Search Configuration.][filtro_cons_texto_filtro] *Text Auto-Complete Field Behavior Interface of the Search Configuration.*

- **Use autocomplete** : The field behaves as an autocomplete according to the values existing in the database.
- **Case Settings** : Allows to convert the letters of the field when losing the focus. The options are:
 - **Upper Case** : All in Upper Case.
 - **Lower Case** : All in Lower Case.
 - **Capitalize first word** : Capitalizes the first letter of the word.
 - **Capitalize all words** : Capitalizes the first letter of all the words.
- **OnChange Submit** : When enabled, it submits the search when there is modifications to the field.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Search Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field.
- **Rows** : Allows to define the amount of records displayed in the field.
- **Width** : Defines the width the size of capture box (Capture Text).
- **Search options** : Allows to define the search settings of the field(Start equal to, Any part and End equals to).
- **Capture Text** : When not enabled, you will only view the field to inform the data. See an example below. *lookup Settings configuration Interface.*
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

 lookup Settings validation configuration Interface.
- **Show label with the description** : Displays a label with the description.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.

- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the Input Object.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
- **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the Input Object.
- **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Search
- 4. Number Auto-Complete

Number Auto-Complete

General Settings

![[Number Auto-Complete Field Behavior Interface of the Search Configuration.]][[filtro_cons_numero_auto]] *Number Auto-Complete Field Behavior Interface of the Search Configuration.*

- **Data Type** : Define the type of field for the application. When set to Number auto complete, you can inform a value based on the select statement of the lookup settings and it will manipulate an internal number for the data.
- **Search Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Use same label used on the Grid** : When this option is enabled, it allows to field to use the same title defined in the grid, in this case the Search Label is not used.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Maximum Size** : Defines the size of the field.
- **SQL Type** : Informs the data type of field in the database.

Field Behavior

![[Number Auto-Complete Field Behavior Interface of the Search Configuration.]][[filtro_cons_decimal_filtro]] *Number Auto-Complete Field Behavior Interface of the Search Configuration.*

- **Use autocomplete** : The field behaves as an autocomplete according to the values existing in the database.
- **Position between values** : This option sets the position that objects will be displayed.
- **Text between values** : Text that will appear between values.
- **OnChange Submit** : When enabled, it submits the search when there is modifications to the field.
- **Show Condition** : When enabled, it displays the condition of the search in the Grid, it will only work if there is at least one option selected.

Values format

![[Number Auto-Complete Field Format of Values with Regional Settings.]][[filtro_cons_decimal_format]] *Number Auto-Complete Field Format of Values with Regional Settings.*

- **Regional Settings** : Allows to apply the Regional Settings to format the number of the field. When not enabled, the attributes Digit Grouping Separator, Decimal Separator, Negative sign and Negative number format.
- **Decimal Precision** : Defines the amount of decimal places for the field.
- **Digit grouping** : Defines if the field will display the digits separator.
- **Decimal Separator** : Defines the decimal separator.
- **Accept** : Determines if the field, will accept only negative, positive or both numbers.

Search Lookup

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field.
- **Rows** : Allows to define the amount of records displayed in the field.
- **Width** : Defines the width the size of capture box (Capture Text).
- **Search options** : Allows to define the search settings of the field(Start equal to, Any part and End equals to).
- **Capture Text** : When not enabled, you will only view the field to inform the data. See an example below. *lookup Settings configuration Interface.*
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

 lookup Settings validation configuration Interface.
- **Show label with the description** : Displays a label with the description.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.

- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
- **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the field.
- **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

This feature allows that the instructions are documented to be used for the generated application, this helps the end user to understand better how the system works.

- **Help Description** : Allows to inform a text that will present to the user when he positions the mouse over the field.
- **Help Type**
 - **Pop-Up** : Selecting the pop-up type, it will display an icon beside the field that when clicked, you will view a pop-up with help description.

Help type - Pop-up configuration Interface.

- **Hint** : Passing the cursor over the field, you will view a hint with the help description.

Help type - Hint configuration Interface.

- **Text** : It will display the help description beside the field.

Help type - Text configuration Interface.

Scriptcase uses the events to enable the developer to customize the application code. Using the events you can program custom actions at a specific execution time (e.g. After a record is inserted, when it is loaded, upon submitting a form, ...) and for a specific application type. In the event areas you can use global and local variables, JavaScript, CSS, HTML, PHP codes and also Scriptcase macros.

onApplicationInit

This event occurs only once, when the application is loaded and before the application executes the SQL. It can be used for data processing or checking variables.

Example: `if ([glo_var_dept] != 'financial'){ sc_redir(app_x.php); }`

onScriptInit

This event occurs when the application scripts are started. It runs before the OnLoad events(when present), however, it will always be executed when running, reloading or navigating the application.

onRefresh

This event runs when the refresh option of the application is enabled.

onSave

This event runs when the application saves the record.

onValidate

This event runs when validating the information of the fields, when submitting the form.

Settings

With this interface, you can set the common attributes of the app.

ATTRIBUTE	VALUE	DESCRIPTION
Application Code	search_products (9.00.0036)	Application code for ScriptCase internal use.
Description	<input type="text"/>	Application description.
Language	English (United States) ▾	Language of the generated application.
Share Location Variable	<input checked="" type="checkbox"/>	If you choose the option YES this application will inherit the locale (Language and Regional Settings) from a global variable, otherwise this application will always load its default locale.
Charset	<input type="text"/>	Application specific charset.
Share Theme Variable	<input checked="" type="checkbox"/>	If you choose the option YES this application will inherit the theme from a global variable, otherwise this application will always load its default schema.
Folder	root ▾	Folder that will store the application on the working project.
Edit by Project	<input checked="" type="checkbox"/>	Allow other users of the same project to edit the application.
Timeout	<input type="text" value="0"/>	Script execution timeout in seconds. Zero uses the PHP default timeout.
HelpCase Link	<input type="text"/>	Associate a HelpCase manual to your application.
Use Enter to	<input type="text"/>	Use the "Enter" key to move from the current field to the next field.

Application Settings Interface

- **Attributes**
 - **Application Code** : It is the name that defines an application. An app can be renamed at the [List of Application](#).
 - **Description** : This field contains a brief description of the application objectives.
 - **Language** : Set the default language of the application. Display all the application hints and messages in the selected language.
 - **Share Location Variable** : Define if the app shares the regional settings with other applications through a session variable.
 - **Charset** : Define a specific charset to use in the application.
 - **Share Theme Variable** : Define if the app shares the Theme settings with other applications through a session variable.
 - **Folder** : Define the project folder that contains the app.
 - **Edit by Project** : Define if other project developers can edit the application.
 - **Timeout** : Set the session runtime timeout in seconds. If the value is Zero, it assumes the default timeout of the PHP.
 - **HelpCase Link** : It allows to associate a [HelpCase](#) file with the application.
 - **Use Enter to** : Allows selecting the action of the Enter key (submit or tabulate the Form).

Error Settings

It groups the notification options of the app.

ATTRIBUTE	VALUE	DESCRIPTION
Show the Error Title in the Application	<input checked="" type="checkbox"/>	Show the title line of the error message in the application.
Show the Error Title in the Field	<input type="checkbox"/>	Show the title line of the error message in the field.
Error Title	{lang_errm_errt}	Title message of the error
Script Error	<input type="checkbox"/>	Display information about the script and line where the error occurred.
SQL Error	<input checked="" type="checkbox"/>	Display the SQL Select Statement that originated the error.
Debug Mode	<input type="checkbox"/>	Run Application on debug mode, showing SQL commands.

Error Settings Interface.

- **Attributes**
 - **Use SweetAlert**: Use the SweetAlert to display messages from the application. When this option is active, it will replace the browser's "confirm" and "alert".
 - **SweetAlert position using Toast** : The position to display error messages on the application.
 - **how the Error Title in the Application** : Define to display the title line of the error message or not.
 - **how the Error Title in the Field** : Define to display the title line of the error message in the field or not.
 - **Script Error** : Allows displaying the line code where there is an error..
 - **SQL Error** : Allows displaying the SQL statement if it got an error.
 - **Debug Mode** : Runs the application in Debug mode, showing all SQL statements the application is executing.

Navigation

This interface allows defining the navigating behavior of the application

ATTRIBUTE	VALUE
Exit URL	<input type="text"/> 🔗
Close on Exit	<input type="checkbox"/>
Redirect URL	<input type="text"/> 🔗
Redirect Variable	<input type="text"/>

Navigation Interface.

Exit URL

URL to where the user goes when he clicks on the “exit” button.

Close on Exit

Close the browser window when the user clicks on the “exit” button.

Redirect URL

Redirect to another URL in case there aren't any global variables available.

Redirect Variable

Creates a variable with the application name and sends it to the redirected application.

Global Variables

This screen shows all global variables used in the application

The global variable is an external parameter required for running the application. The application can include global variables in the WHERE clause, field definitions and names, event programming, and so on.

You can define global variables by using square brackets ([variable]). You can pass the parameters to the application through one of the methods: Session, Post, and Get

if you need to use [Database Identifiers](#) in or tables, we recommend using the double quotation marks(") instead of the brackets([]) because of the conflict with the [Global Variables syntax](#) of Scriptcase. Using the brackets as database identifiers may lead to problems in the Scriptcase applications.

- ⚙ Settings
- 🔗 Navigation
- 📄 Messages
- 🌐 Global Variable

Global variables Interface.

Example:

```
Select CustomerID, CustomerName from Customers WHERE CustomerID = '[v_customerid]'
```

We do not recommend using session variables (global variables) with the same name as the table fields.

In this case, the variable **v_customerid** is displaying in the global variables configuration.

ATTRIBUTE	VALUE
global	<div style="border: 1px solid #ccc; padding: 5px;"> <p>Scope</p> <p><input type="checkbox"/> SESSION</p> <p><input checked="" type="checkbox"/> POST</p> <p><input checked="" type="checkbox"/> GET</p> <hr/> <p>Settings</p> <p><input type="checkbox"/> Optional</p> <hr/> <p>Type</p> <p><input type="radio"/> Out</p> <p><input checked="" type="radio"/> In</p> </div>

Global variables configuration Interface.

Attribute

It shows the variable's name in the application.

Value

Defines the behavior of the variables, divided into three blocks, they are:

Scope

Defines how the application receives the variable. A variable defined as the POST method in the app does not receive value if it comes with the GET method. Those are the methods available:

- **ession**: Defines that the variable must be created as a PHP session variable by another application.
- **ET**: Defines that the variable must be passed by the URL, that is, visible by the browser.
- **OST**: Defines that the variable must be passed through the POST method of an HTML form.

Settings

- **ptional**: Check it to don't validate the variable at runtime.

Type (In / Out)

Defines whether the variable comes from another application (In), or it is starting in the current app (Out).

Description

Indicates where the application is using the variable.



1. [Home](#)
2. Applications

General Overview

Allows to create link between applications that are in the same project, the link option incorporates various functionalities like above.

New Link

To create a new link you just need to access the option **Link between Applications** located in the menu at the left side.

When clicking in **New Link** the screen below will be display:

When clicking in **New Link** the screen to choose the type of link that will be created will be displayed.

[Edit Link:](#)

Allows to edit registers of the grid using a form.

[Capture Link:](#)

Allows to create a link specifically to the filter's fields, capturing data from other grid.

[Field Link:](#)

Allows to create a link represented through a link, from a grid's column to any application from the project.

[Button Link:](#)

Allows to create a link through grids buttons to any application from the project.

1. [Home](#)
2. [Applications](#)
3. Search
4. Application Link

Application Link

Creating an Application Link

This type of link allows the developer to create a link from a grid to a form with the objective of editing the register of a grid's row.

In the link options, we will choose the **Application Link**. When we choose this option, edit a register from a Grid will be possible.

Applications list

After we select this option, will be displayed the applications list to what you want to create the link.

This screen can be viewed from the following ways:

All:

In which is possible to see all the project's applications. **Example:**

By folder:

In which is possible to see the applications according to the folders in which each of them are.

By type:

In which is possible to see the applications grouped by its respective types.

Definition of Parameters

In this setting we are going to define the type of parameter that will be passed to the next application.

We have two options at **Type of Parameters**, they are:

Fixed value:

This option allows the user to define a fixed value that will be used to the call of the next application.

No value:

This option allows to create the link without the need to send any parameter to the next application.

It is also possible to see a refresh button at its right side.

This button should be used when a new parameter is added to the target application, so the new parameter can be loaded in the current application to make the link.

Link properties

In this screen we will set the application display mode that will be called in the link.

There are five display options, they are:

Open in the same window:

When we use this option the target application will be displayed in the same window of the application, and the target application will have a back button so we can return to the previous application.

Open in another window:

When we use this option the target application will be displayed in other browser window, and the target application will have a exit button so we can close this window.

Open in an iframe:

When we use this option the target application will be displayed in the same window of our application, however it will be displayed in a iframe in one of the four options available in the iframe settings.

Open in a parent:

When we use this option the target application will be displayed in the same window of our application, and the target application will have a back button so we can return to the previous application.

Modal:

When we use this option the target application will be displayed in the same window of our application, however it will be displayed in a small window above the current application.

Display the button new in the grid:

When we select the option **Yes**, the button **New** is created in the grid to insert new registers.

Also, when we select the option **Yes**, it is displayed two new options **Label to the New button** and **Hint to the New button**:

Label to the New button:

In this option we will inform a name for the button that will be displayed in the Grid.

Hint to the New button:

In this option we will inform a message that will be displayed when the mouse cursor is over the **New** button.

Shortcut key to the New button:

Indicates the shortcut key to the button add new register.

Display the button New in the grid:

When we select the option **Yes**, the button **New** is created in the grid to insert new registers.

Shortcut key to the New button:

Indicates the shortcut key to the button add New Register.

Label to the New button:

In this option we will inform a name for the button that will be displayed in the Grid.

Hint to the New button:

In this option we will inform a message that will be displayed when the mouse cursor is over the **New** button.

Open in an iframe.

When we use this option the target application will be displayed in the same window of our application, and the target application will have a back button so we can return to the previous application.

When we select this option, we can set the following options:

Shortcut key to the New button:

Indicates the shortcut key to the button add New Register.

Exit URL for the target application:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Iframe properties

In this settings, we can define some iframe details that will display the target application.

Display the target application's header:

When the option **Yes** is selected, the header of the target application is also displayed in the iframe.

Iframe position in relation to the main application:

In this option, we will define in which position in relation to the main application the iframe will be displayed, there are four options:

Below: The iframe will be displayed below the main application.

Above: The iframe will be displayed above the main application.

Right: The iframe will be displayed on the right of the main application.

Left: The iframe will be displayed on the left the main application.

Action after an insert:

In this option, we will define what will happen after a register insert, there are two options:

Reload the grid: The current page will be refreshed after the insert.

Move to the end of the grid: After the insert will be displayed the last page of the grid will the last register inserted.

Iframe height:

Allows to set the iframe height.

Iframe width:

Allows to set the iframe width.

Display the button New in the grid:

When we select the option **Yes**, the button **New** is created in the grid to insert new registers.

Shortcut key to the New button:

Indicates the shortcut key to the button add New Register.

Label to the New button:

In this option we will inform a name for the button that will be displayed in the Grid.

Hint to the New button:

In this option we will inform a message that will be displayed when the mouse cursor is over the **New** button.

Form Properties

In those settings, we can define which buttons will be available in the target form application. Initially we have five options, that are:

Enable Insert button:

In this option we can define if the **New** button will be available in the target application.

Enable Update button:

In this option we can define if the **Save** button will be available in the target application.

Enable Delete button:

In this option we can define if the **Delete** button will be available in the target application.

Enable Navigation buttons:

In this option we can define if the buttons **first, previous, next and last** will be available in the target application.

Enable register editing button in the grid:

In this option we can define if the edit register button, which is the **pencil** in the grid, will be available.

1. [Home](#)
2. [Applications](#)
3. Search
4. Capture Link

Capture Link

Applications list

After we select this option, will be displayed the applications list to what you want to create the link.

This screen can be viewed from the following ways:

All:

In which is possible to see all the project's applications. **Example:**

By folder:

In which is possible to see the applications according to the folders in which each of them are.

By type:

In which is possible to see the applications grouped by its respective types.

Definition of Parameters

In this setting we are going to define the type of parameter that will be passed to the next application.

We have two options at **Type of Parameters**, they are:

Fixed value:

This option allows the user to define a fixed value that will be used to the call of the next application.

No value:

This option allows to create the link without the need to send any parameter to the next application.

It is also possible to see a refresh button at its right side.

This button should be used when a new parameter is added to the target application, so the new parameter can be loaded in the current application to make the link.

Application with no parameters

However, when the target application does not have any defined parameters, the following screen is displayed:

![Without parameters][semparametro]

Clicking in the button, you will be taken to the target application to create a parameter, so you can use the update button in the **parameters definitions** to refresh them.

Link properties

In this screen we will set the application display mode that will be called in the link.

In this type of link there is only one display option:

Modal:

When we use this option the target application will be displayed in the same window of our application, however it will be displayed in a small window above the current application.

Modal

When we use this option the target application will be displayed in the same window of the application, however she will be displayed in a small window above the current application.

When we select this option, we can set the following options:

Modal:

In this option we are going to define if the target application will be opened in a Modal.

Yes: This option will make the target application be opened in a modal. **No:** This option will make the target application be opened in a new window.

If **Yes** is selected in the previous option, the Modal **Height** and **Width** will be available.

Height:

Allows to set the Modal height.

Width:

Allows to set the Modal width.

If **No** is selected in the previous option, only those options will be available.

Allows to modify manually in the update:

When we select the option **Yes**, the button **New** is created in the grid to insert new registers.

Allows to modify manually in the insert:

Indicates which shortcut key to the button add new register.

Allows to modify automatically in the update:

Exit URL to the called form. If any value is defined, the return address (back button) will be the exit URL of the called form.

Allows to modify automatically in the insert:

Choosing the option **Yes**, the window will be closed after doing the insert of a register.



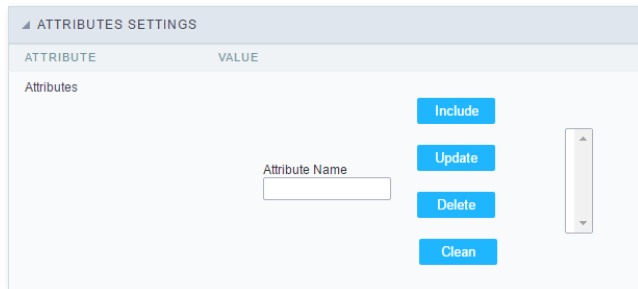
1. [Home](#)
2. [Applications](#)
3. Search
4. Button link

Button link

ScriptCase has incorporated the concept of Object Oriented programming, using attributes, resources, methods and libraries. It is possible to create your own business rules in applications, and by using these concepts you can reap huge rewards in terms of better organization and improved development.

Attributes

The attributes are variables that has the global scope declared in the application. An attribute that can be referenced in all the methods and the application's events.

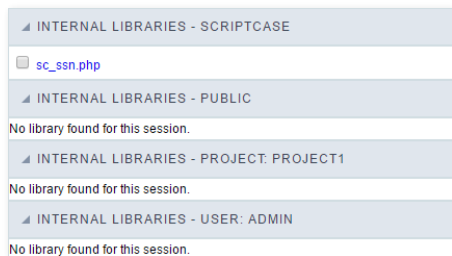


Attributes configuration Interface.

- **attributes**
 - **attribute Name** : This field allows to inform the attributes name, after typing the name just click on **include**. In case you want to **update** click on the attributes name (located on the buttons' right side), do the necessary modifications and click on update. In case it is necessary to **delete** an attribute, it is necessary to do same process as the update, but when selecting the attribute, it is necessary to click on the delete button. The **clear** button, as the name suggests, it clears the list of all the attributes listed on the right.

Internal libraries

To use the internal libraries just select the correspondent (Project Libraries for example). This process allows the methods existing in the library, to be visible in all the application's Events and Methods. See how to manage the libraries by [clicking here](#).

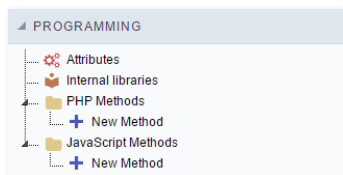


Internal Libraries management Interface

PHP Methods

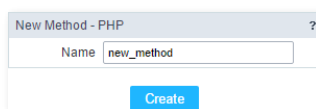
Methods are function or procedures declared by the developer, that helps when applying the routines. Using methods in the applications allows to reuse your code throughout the application, optimizing the development experience.

Creating A New Method



- Give the method a name and click Create. As in the image below.

PHP methods are incorporated into the same class, so it is not possible to use two functions with the same name even when used in different methods.



- Methods can receive parameters.

```
function new_method
1 echo "Hello World!!!";
```

- dd the amount of variables:

Definition of the parameters of the method:

new_method

- No defined parameter.

Add Parameter(s) Cancel

Save

- efining the variables:

Insertion of Parameters

Name	Type	Value Standard
<input type="text"/>	For Value	<input type="text"/>
	For Value	
	For References	

Save Back Cancel

- **ame** : Type in the variable's name.
- **ype** : Selecting the type of variables: For Value or For Reference.
- **alue Standard** : The parameter's value used to initialize when calling the method.

References allows to create a second name for a variable that you can use to read and modify the variable's original information.

- diting a parameter:

Definition of the parameters of the method:

new_method

Parameters

Stest = test

⏪ ⏩

☑ ☐ ✎ ✕

Add Parameter(s) Cancel

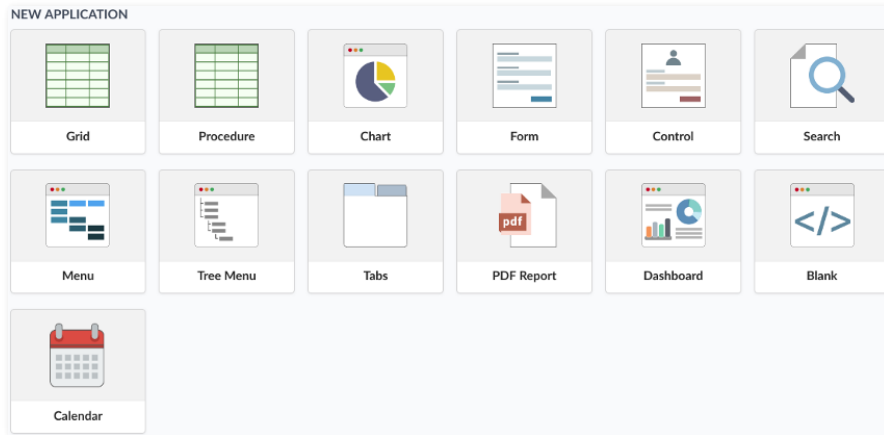
Save

- ☑ : Checking all the parameters in the parameter list.
- ☐ : Uncheck all the parameters in the parameter list.
- ✎ : Edit the selected parameter of the list.
- ✕ : Deletes the selected variable of the list.

New Application

The creation process of one application is simple enough and intuitive.

You just need to select which application you want to generate in the applications available in Scriptcase, That it will be generated automatically.



Application Data

When the application is selected, the following screen will be displayed below.

In this table you need to select the connection that will be used in the application. The tables will be loaded and displayed in the field **Table** when the connection is selected.

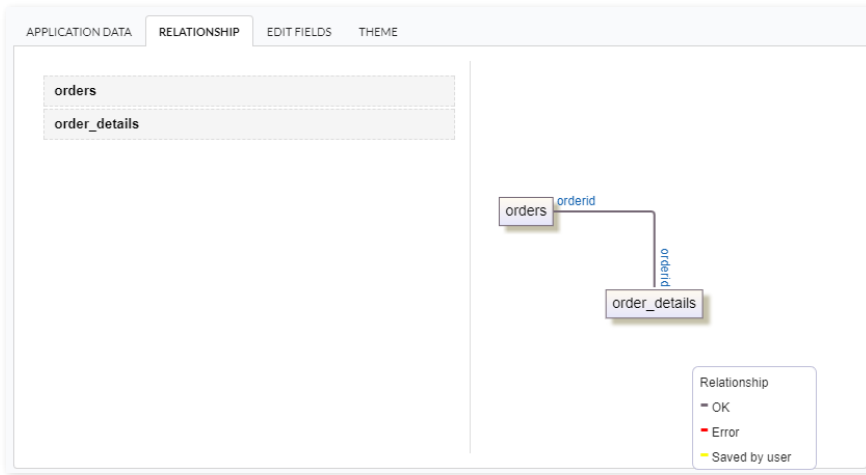
It is possible to select more than one table in the application creation.

After selecting the table and fields that should be part of the application, the name of the application will be formed by the type of application + name of the table. **Ex: grid_orders** However it is possible to change the application's name to any other desired name.

- **connection** - It defines the used connection to create the application. All the project's application will be displayed.
- **ables** - It defines the used tables in the application. (Form and Calendar can only use one table).
- **ields** - It defines the fields that will be part of the applications.
- **QL Select Statement** - Display the select statement created after selecting the tables and fields. This field allows also to insert a previously created SQL, if the used tables are available in the database selected in the connection.

Relationship

When select two or more tables, the tab "Relationship" will be displayed. In this tab we can see the relationship created between the tables, where we can edit the related fields.



When we click in a link, in the screen above, it will displayed the related field's edition form, as you can see in the image below.

Edit Fields

In this tab will be displayed the fields that are going to be created in the application, also its labels and datatypes. We can make define some settings here like changing its labels and datatypes.

Fields	Label	Datatype	Search
id	Id	Integer	<input checked="" type="checkbox"/>
title	Title	Text	<input checked="" type="checkbox"/>
description	Description	Text	<input checked="" type="checkbox"/>

- **ields** - Name of fields stored in the database.
- **abel** - Name of fields displayed in the generated application.
- **atatype** - Field's datatype.
- **earch** - It defines the available fields in the search.

Theme

In this screen you select the theme that the application will use. The default project's theme, that can be checked and changed in **Project > Properties**, will be selected automatically.

- **heme** - List of available themes in the project.

After all the settings are done, click in the button **Create**. In the next screen the settings of the created application will be displayed.

PDF Report Settings

PDF REPORT SETTINGS	
ATTRIBUTE	VALUE
PDF Orientation	Portrait ▼
Type	Records per Page ▼
Friendly URL	<input type="text"/>
PDF measure unit	Millimeters ▼
PDF Format	Letter (216 x 279 mm) ▼
PDF Destination	Browser ▼
Automatic page break	<input checked="" type="checkbox"/>
Upper Margin	<input type="text"/>
Bottom Margin	<input type="text"/>
Right Margin	<input type="text"/>
Left Margin	<input type="text"/>
Search Create	<input type="checkbox"/>
Page Amount	<input type="text" value="1"/>
Amount of Columns	<input type="text" value="1"/>
Columns Width	<input type="text" value="0"/>
Columns Height	<input type="text" value="0"/>



PDF Report Settings Interface.

- **Attributes**

- **PDF Orientation** : Using this option you can set whether to print the PDF in portrait or landscape orientation.
- **PDF type** : PDF type
 - **records per Page** : Only applicable to Grids aligned horizontally, which allows you to set the number of records that will be printed per page.
 - **records per Line** : Allows you to set the records that will be printed per page in the pdf.
- **Friendly URL** : This field allows you to change the URL that will be called by the application. Allowed characters are the same available on URLs: a-z, A-Z, 0-9, -_. This option can also be changed on the home screen, on the "Friendly URL" column at the applications list.
- **PDF measure unit** : Using this option you can define the unit of measurement of the position of objects in the PDF. Used in printing format, it can be: points, millimeters, centimeters and counts.
- **PDF Format** : Using this option you can select the Form type in which the PDF will be printed (letter, A4, etc).
- **PDF Destination** : Using this option you can set where the PDF will be sent to the Browser, Download or server.
 - **browser** : If you use this option the PDF will open within the browser.
 - **download** : The PDF will open for download. you must inform the name of the file to be generated. E.g.: file.pdf
 - **server** : The PDF will be sent to the server. It is necessary to inform the path of the file to be generated. E.g.: c:\folder\file.pdf
- **Automatic page break** : Use automatic page break on pages.
- **Margins** : Allows you to view the application on the page according to the values reported in the margins (right, left, top and bottom) in millimeters.
- **Search Creation** : Using this option you can set the PDF startup mode by the Search. So the initial mode will be a Search application.
- **Page Amount** : Allows you to set the number of pages that will be printed in the PDF.
- **Amount of Columns** : Allows you to set the number of columns that are printed, what allows you to create applications for issuing of labels printing for example.
- **Columns Width** : Allows you to set the width of the columns in the application.
- **Columns Height** : Allows you to set the height of the columns in the application.

Font and Background

Configuring the PDF Preview allows you to set values of text and layout preview.

Font and Background	
ATTRIBUTE	VALUE
Text Font	<input type="text" value=""/>
Text Font Size	<input type="text" value="12"/>
Font Color	<input type="text" value="#000000"/> 
Text Font Style	<input type="checkbox"/> Bold <input type="checkbox"/> <i>Italic</i> <input type="checkbox"/> <u>Underline</u>
Show Ruler	<input type="checkbox"/>
Image	<input type="text" value=""/> 
Width	<input type="text" value="0"/>
Height	<input type="text" value="0"/>
X Position	<input type="text" value="0.000000"/>
Y Position	<input type="text" value="0.000000"/>
To print	<input checked="" type="checkbox"/>

PDF Report visualization interface settings.

- **ext Font** : Here you can choose the font family to be used initially by the Report PDF.
- **ext Font Size** : Here you can choose the font size .
- **ont Color** : Here you can choose the font color .
- **ext Font Style** : Allows you to choose the style of the text, if this will be bold, italic, or underlined, or any other combination.
- **how Ruler** : Sets whether the PDF printing displays ruler at the edges of the pages.
- **mage** : Allows you to insert a background image on the page. This image will be also printed, it can be a watermark for your document for example.
- **idth** :Sets the image width, in pixels, to be inserted into the page.
- **eight** :Sets the image height, in pixels, to be inserted into the page.
- **Position** : Sets the abscissa of the initial image position.
- **Position** : Sets the ordinate of the initial image position.
- **rint** : This option sets whether the image will be printed on the application.

SQL Settings

This interface allows configuring the related database settings, such as the SQL statement, the used database connection, case sensitive, and others.

ATTRIBUTE	VALUE	DESCRIPTION
SQL Select Statement	<pre>SELECT customerid, companyname, contactname, contacttitle, birthdate, country, regionid, stateid, city,</pre>	
Limit	<input type="text"/>	It sets the number of records to be retrieved from the SQL statement.
SQL Preparation	<input type="text"/>	
Connection	<input type="text" value="conn_example"/>	Connection name to access the database.
Use Customized Message	<input type="checkbox"/>	Use a customized error message when the application has no records.
No Records Message	<input type="text"/>	When the application has no records, it will display this customized text.
Font	<input type="text" value="Aa"/>	Font face of the error message.
Font Size	<input type="text" value="12"/>	Font size of the error message.
Font Color	<input type="text" value="#000000"/>	Font color of the error message.
Variable for Table	<input type="text"/>	Variable name used for replacing the table name. Please indicate the name of the table that will be replaced by the variable value.
Fields Variables	<div style="border: 1px solid black; padding: 2px;"> <p>Variable</p> <input type="text"/> <p>customerid</p> </div>	Variables for substitution of the field names on the application. For each dynamically determined field, inform the name of the variable and the field that will be substituted.
Case Sensitive	<input checked="" type="checkbox"/>	Use case sensitive.

Grid SQL configuration

SQL Select Statement

It allows you to define the primary SQL of the application. You can edit this SQL to add or delete fields.

Limit

Lets you limit the display in the number of records retrieved by SQL query.

SQL Preparation

You can enter SQL commands or procedure names to execute them before the primary SQL of the application.

Connection

It allows defining the database connection of the application. You can change the connection to another one that has the same table.

Use Customized Message

Lets you define to display the "no records" message or not.

No Records Message

Lets you set the message when the application has no records.

Font

This option is available when using the "Use Customized Message" option. It lets you set the font for the message.

Font Size

This option is available when using the "Use Customized Message" option. It lets you to set the font size.

Font Color

This option is available when using the "Use Customized Message" option. It lets you to set the font color.

Variable for Table

It allows to use a variable to change a part of the string containing the table name.

Fill the first input with the name of the variable (replace to). The second input you should fill with the part of the name of the table you want to replace (replace from).

Fields Variables

It allows to use a variable to change a part of the string containing the field name.

Fill the first input with the name of the variable (replace to). The second input you should fill with the part of the name of the field you want to replace (replace from).

Case sensitive

It defines if the database connection uses case sensitive or not.

Security ?		
ATTRIBUTE	VALUE	DESCRIPTION
Use Security	<input type="checkbox"/>	Application needs validation to run. Read about the sc_apl_status macro.
Url output of the security	<input type="text"/>	Uri to be redirected if the user does not have access to this application.
Use Password	<input type="checkbox"/>	Enter password required to run the application.
Request password just once	<input type="checkbox"/>	Request password just once per session without asking for password every time you open the application in the same session.
Enable direct call by URL	<input checked="" type="checkbox"/>	Allow an application to be called by typing the URL directly in the browsers.

Application Security Configuration

Use Security

Turns the application accessible just when using the macro sc_apl_status or using the security module. When it is enabled, the application displays a message "User not Authorized" if the user tries to access it directly.

Url output of the security

It redirects the application to another URL after displays the warning "User not Authorized".

Use Password

Turns the application accessible just by using a password.

Request password just once

Request the password once per session, not needing to inform the password every time you access the application.

Enable direct call by URL

Allows to call the application directly from the URL in the browsers.

Allows calling the application directly from the URL in the browsers. In the development environment, you can disable those options for testing, it avoids requiring the login every time you run the application. To disable the security and password in development, you must access Options > My ScriptCase and uncheck the options: Enable Use of Security, Enable Use of Password.

This interface allows you to define a Log schema to the app. The Log schema tracks what the end-users are doing in the app. Those are the events it can tracks: Access, Insert, Update, and delete.

To learn how to create a Log schema [see here](#).

The screenshot shows a configuration window titled "LOG". It features a table with two columns: "ATTRIBUTE" and "VALUE". The "Schema" attribute has a dropdown menu currently set to "log". Below the table, there are two empty list boxes: "Events" on the left and "Access" on the right. Between these two boxes are two blue buttons: "Add >>" and "Remove <<".

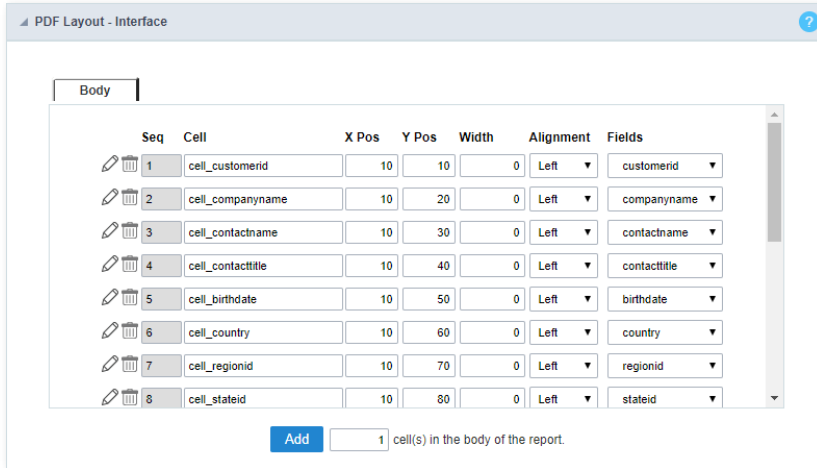
Application Log configuration

Schema Select an existing log schema for the application.

Events You can select just specific events to save in the log.

Interface

Using this interface you can define the display layout of cells containing the values of the fields in the body of the report.



PDF interface settings.

- **tributes**
 - o **eq** : Sequential number of cells (ascending order).
 - o **ell** : Cell name.
 - o **os X** : Sets the abscissa of the cell.
 - o **os Y** : Sets the ordinate of the cell.
 - o **idth** : Sets the cell width.
 - o **lignment** : Sets the cell alignment.
 - o **ield** : Here you have to select the field that will be displayed within the cell, according to the application SQL.

Code

Scriptcase creates the codes automatically, so by changing that within the option "code" you will assume the PDF creation PHP code.



Configuration interface of the PDF Code.

The application "Report PDF" was developed based on a library called [TCPDF](#), so that you can use the available library methods or a corresponding macro from the table below. In order to use some method you must use the \$pdf object, for example: \$pdf->AcceptPageBreak(parameters).

In order to access the TCPDF documentation [Click here](#)

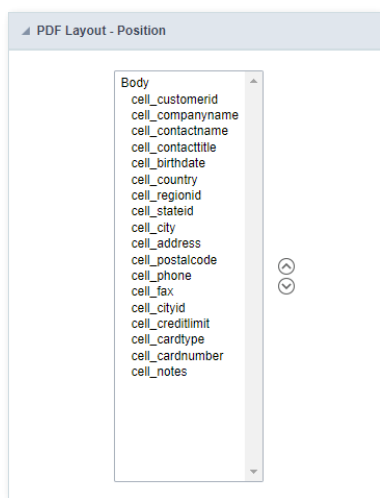
PDF Report Macros

TCPDF Method	Scriptcase Macro	Description
AcceptPageBreak	sc_pdf_accept_page_break	Accept automatic page break.
AddFont	sc_pdf_add_font	Add a new font
AddLink	sc_pdf_add_link	Creates an internal link
AddPage	sc_pdf_add_page	Add a new page
AliasNbPages	sc_pdf_alias_nb_pages	Defines an alias for the number of pages
Cell	sc_pdf_cell	Prints a cell
Close	sc_pdf_close:	Ends the document

PDF Method	Script Case Macro	Description
Footer	sc_pdf_footer	Prints a footer message
GetStringWidth	sc_pdf_get_string_length	Calculates the size of a string
GetX	sc_pdf_get_x	Returns the current x position
GetY	sc_pdf_get_y	Returns the current y position
Header	sc_pdf_header	Page header
Image	sc_pdf_image	Places an image on the page
Line	sc_pdf_line	Draws a line
Link	sc_pdf_link	Enter a link
Ln	sc_pdf_ln	Configure the line break
MultiCell	sc_pdf_multi_cell	Prints a text with line breaks
Output	sc_pdf_output	Save or send the document
AliasNumPage	sc_pdf_page	Returns the current page number
AliasNbPages	sc_pdf_page_tot	Returns the total number of pages
Rect	sc_pdf_rect	Draws a rectangle
SetAuthor	sc_pdf_set_author	Defines the author of the document
SetAutoPageBreak	sc_pdf_set_auto_page_break	Enables or disables the automatic page break mode
SetCompression	sc_pdf_set_compression	Page compression on/off
SetCreator	sc_pdf_set_creator	Sets the document's creator.
SetDisplayMode	sc_pdf_set_display_mode	Toggles the display mode
SetDrawColor	sc_pdf_set_draw_color	Sets the color used for drawing operations
SetFillColor	sc_pdf_set_fill_color	Sets the color for fill operations
SetFont	sc_pdf_set_font	Sets the font settings
SetFontSize	sc_pdf_set_font_size	Sets the font size
SetKeywords	sc_pdf_set_keywords	Associates a key word to a document
SetLeftMargin	sc_pdf_set_left_margin	Sets the left margin
SetLineWidth	sc_pdf_set_line_width	Sets the width of the line
SetLink	sc_pdf_set_link	Sets the target of an internal link
SetMargins	sc_pdf_set_margins	Sets the margins
SetRightMargin	sc_pdf_set_right_margin	Sets the right margin
SetSubject	sc_pdf_set_subject	Sets the subject
SetTextColor	sc_pdf_set_text_color	Sets the color of the text
SetTitle	sc_pdf_set_title	Sets the title of the document
SetTopMargin	sc_pdf_set_top_margin	Sets the top margin
SetX	sc_pdf_set_x	Sets the current position of x
SetXY	sc_pdf_set_xy	Sets the current position of x and y
SetY	sc_pdf_set_y	Sets the current position of y
Text	sc_pdf_text	Prints a string
Write	sc_pdf_write	Prints a string of characters

Positioning

This interface allows you to define the display order of the fields (selecting through the arrows next to the right frame).



Configuration interface of the PDF fields placement.

1. [Home](#)
2. Applications

General Overview

The applications can have two types of fields.

Table Fields

Columns of the table connected to the application and their data types.

Example:

In a table with the following structure:

```
CREATE TABLE `orders` (
  `orderid` int(11) NOT NULL,
  `customerid` varchar(5) NULL DEFAULT NULL,
  `employeeid` int(11) NULL DEFAULT NULL,
  `orderdate` date NULL DEFAULT NULL,
  `requireddate` date NULL DEFAULT NULL,
  PRIMARY KEY (`orderid`) USING BTREE
);
```

These are the listed fields on left menu **Fields**:

Virtual Fields

Fields created inside application to assist in the development process.

How to create a virtual field

1. Inside an application, access the left menu **Fields** and click on **New Field** option.
2. Choose how many fields do you want to create:

1. Create the fields and define type, name and label for each one:

- **Type:** Data type of the field.
- **Name:** Internal name of the field. Used to identify the field on events and Scriptcase interface.
- **Label:** Title of the field displayed on the application.

Scriptcase **doesn't insert** virtual fields on the table after its creation.

Types of fields by applications

Application Table Fields Virtual Fields

Grid	<input type="text"/>	<input type="text"/>
Procedure	<input type="text"/>	<input type="text"/>
Chart	<input type="text"/>	<input type="text"/>
Form	<input type="text"/>	<input type="text"/>
Control	<input type="text"/>	<input type="text"/>
Search	<input type="text"/>	<input type="text"/>
PDF Report	<input type="text"/>	<input type="text"/>
Calendar	<input type="text"/>	<input type="text"/>

The applications **Menu**, **Tree Menu**, **Dashboard** and **Blank** were not listed because they haven't manage fields.

- 1. [Home](#)
- 2. [Applications](#)
- 3. [ReportPDF](#)
- 4. [Text](#)

Text

General Settings

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced by the curly brackets {}. While running the application, the field within the curly brackets is replaced by its value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in :**
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Martial Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

1. [Home](#)
2. [Applications](#)
3. ReportPDF
4. Multiple Lines Text

Multiple Lines Text

General Settings

Multiple Lines Text field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Multiple Lines Text, it accepts letters, numbers and special characters in multiple lines.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Case Settings** : Convert the letter from the field when losing focus. The options are:
 - **Upper case** : All in Upper Case
 - **Lower case** : All in Lower case
 - **Capitalize first word** : Capitalizes the first letter of the first word
 - **Capitalize all words** : Capitalizes the first letter of all the the words
- **Show HTML content** : Determines if the HTML contained in the field will be displayed or not. If enabled, the HTML will be displayed, otherwise the HTML will be interpreted by the browser.
- **Grid Mask** : Defines the mask for the field display. There are two typed of masks described below:

Character	Description
X	Placeholder to any character. Replaced by any character. If number of characters entered are less then the mask size, the field value is completed with zeros (Filling full size field entry is required).
Z	Replaced by any character retrieved from database. Suppress zeros at field left (Complete field filling is optional). When used combined with the mask character X it should be placed at the mask left.

Mask Examples

Mask	Field Value	Formatted Value
(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
(zz) xxxx - xxxx	1234567890	(12) 3456 - 7890
(zz) xxxx - xxxx	12345678	() 1234 - 5678
(zz) xxxx - xxxx	0012345678	() 1234 - 5678

- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.

- **Run content in JavaScript** : If enabled, the JavaScript will be interpreted by the browser, otherwise the JavaScript will be displayed.
- **SQL Type** : Informs the data type of field in the database.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.

- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
- **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
- **Summary** : Applies the lookup only for the summary (HTML and PDF).
- **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

1. [Home](#)
2. [Applications](#)
3. [ReportPDF](#)
4. [Integer](#)

Integer

On this page, the developer can configure the settings for your field of type number. From the use of specific display symbols to the mode in which they are displayed. Thus, it can streamline its application.

Data type:

It sets the application field type. When the field is set as Number, it is permissible to define formatting rules of integers.

Field Mask Input:

It sets the mask input for the field. There are two types of mask as described in the table below:

Character Description

X	It is overridden by any characters returned by the database. Filled is required and the value will be completed with leading zeros when there are fewer characters than the size of the mask.
Z	It is overridden by any characters returned by the database. Its filling is optional and when there are fewer characters than the mask size, nothing will be done about the characters that are missing. In addition the leading zeros will be suppressed. When used in conjunction with the X necessarily the use of this character is to the left of the mask.
9	It represents any numeric character (from 0-9)
A	It represents an alpha numeric character (A-Z,a-z)
*	It represents any alpha-numeric character (A-Z,a-z,0-9)entered by the user.

Mask Input Examples:

Field	Mask Input	Typed Value	Formatted Value
Phone Number	+99 99 9999 - 9999	123456789012	+12 34 5678 - 9012
Phone Number	(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
Phone Number	(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
Phone Number	(zz) xxxx - xxxx	12345678	() 1234 - 5678
Software Key (Letters only)	AAAA-AAAA-AAAA-AAAA	QWERTYUIASDFGHJK	QWER-TYUI-ASDF-GHJK
Software Key (Letters and numbers)	xxxx-xxxx-xxxx-xxxx	Q1W2E3R4T5Y6U7I8	Q1W2-E3R4-T5Y6-U7I8
License Plate	AAA - 9999	QWE1234	QWE - 1234
Scriptcase Serial Key	A999A999A999-**	D111H111G111DG2P	D111H111G111-DG2P
Multiple masks (Phone Number)	9999-9999;(99)9999-9999; 9999 999 9999		+99 99 9999-9999

Tipo SQL:

Reports the SQL type of the field as it is configured in the database.

Use regional settings:

It allows you to apply regional settings to the number fields formatting. When this option is not selected it will be displayed the attributes group, minus sign and negative number format.

Grouping:

It allows you to define which character is used to separate thousands. This option is only available when the option to use regional settings is disabled.

Negative sign:

It allows you to define which character will be used for the display of negative numbers. This option is only available when the option to use regional settings is disabled.

Negative number format:

It allows you to define the placement of the negative sign in relation to value. This option is only available when the option to use regional settings is disabled.

Color for the negative values:

It allows you to define a color when the value is negative, improving the understanding of the end user about that kind of value.

Example:

Display the value in words:

The value of the field will be displayed in full on application. This feature can facilitate the comprehension and understanding of the user.

Example:

Line size:

Maximum size in characters to be displayed in the value cell, in full. When this value is exceeded the line will break within the cell.

Lookup Settings

The Lookup is a way used to provide the end user a list of values description/name where the items can be selected and used in the application. For example, for a field sex that in the database is represented with M or F values can be presented as male or female dynamically with the lookup. So you will be able to retrieve these values from a database table or manually, if the reference table does not exist.

Example:

Lookup Methods

There are two lookup methods available:

Manual: To use this method it is necessary to inform the selection conditions manually to the Lookup.

Automatic: To use this method it is necessary to use a lookup query from the database to access the reference table.

Editing-automatic lookup:

Select Command:

It sets the SQL command that will retrieve the value to display inside the Grid field. The command must have the following format

```
SELECT field_to_be_displayed FROM table WHERE key_field = {grid_field}
```

The Grid field must be always referenced between curly braces {}, So at run time, the field braces will be replaced by its value in the table.

Multiple Options:

When the Select command informed return multiple values, the option must be selected.

Delimiter:

It sets the tab of values returned by the Select command, this option should be filled when the "Yes" option is checked in the multiple attribute options.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

Use a specific connection:

This option allows you to select another existing connection in the project, other than the current connection, to select the values for the lookup.

Edit Lookup - Manual:

Lookup used to modify the display of the field with predefined values.

Lookup Types

It sets the operating mode of the lookup can be: single value, multiple values (delimiter), multiple values (position) and multiple values (binary).

Single value: it will be displayed only one value for the field. There is a direct replacement of the value stored in the Bank for a label defined in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value of "M" will be replaced by "Male".

Multiple Values (Delimiter):

By using this option it will be displayed multiple values for the selected field. The values must be separated by a delimiter to be informed. A replacement of parts from a string stored in the field, separated by a delimiter for values contained in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value S;M will be replaced by Sports and Movies.

Delimiter: Character used to separate the values within the same string.

Multiple Values (Position)

Allows you to recover information stored in a single string of the selected field. In order for this information to be retrieved must be informed, in addition to the label, the starting position and the number of bytes that each information occupies within the string.

As an example we will use a string to store Sex, Marital status and Hobby respectively. Sex occupies one byte, Status one byte and Hobby two bytes .

For this purpose we define the list as:

Label	Value	Start	Size
Male	M	1	1
Female	F	1	1
Married	M	2	1
Single	S	2	1
Study	ST	3	2
Sportes	SP	3	2
Reading	LE	3	2

Example: the string MMST would be presented in the query as: Male Married Study

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value M will be replaced by Male.

Start: : Starting position in the string where the information is recorded. The first position is always 1.

Size: : Number of bytes that the information occupies in the string.

Multiple values (binary):

It allows you to retrieve several information stored in decimal form of the selected field.

As examples we will use the following list (although not informed in the interface of inclusion of values in the list, each item has a value assigned automatically according to the order):

Assigned value	Description in Lookup
1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

To display the data, a decomposition of the decimal number stored in the bank is performed. For example, numbers 11 and 12 (written in the database) would be decomposed as follows:

$$11 = 1 + 2 + 8 = (\text{Sports} - \text{Culture} - \text{Reading})$$

Assigned value	Description in Lookup
1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

$$12 = 4 + 8 = (\text{Leisure} - \text{Reading})$$

Assigned value	Description in Lookup
1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

Label:

Text that will be displayed in the Grid.

Include Button:

Includes in the list the values populated in the Label and value fields.

Alter Button:

It changes the attributes of the selected item.

Delete Button:

Deletes the selected item.

Clean Button:

It cleans the fields

Save Button:

Allows the user to save the entire list, for later use using the Load Lookup definition option.

Load Lookup definition:

Allows you to load a predefined value list for use as a lookup.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

1. [Home](#)
2. [Applications](#)
3. ReportPDF
4. Decimal

Decimal

General Settings

Decimal field Configuration Interface.

- **Data Type** : Select the type of field for the application. When it is defined as a Decimal you can define the format of a decimal number.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Accumulated** : When enabled, define that the field will be accumulated with the values of another field. This option is only available on virtual fields.
- **Accumulator field** : Only available when the option "Accumulated" is enabled. This configuration defines which field will be used to accumulate the values. The chosen fields needs to be Integer, Currency or Decimal.
- **Grid Mask** : Defines the mask for the field display. There are two typed of masks described below:

Character	Description
X	Placeholder to any character. Replaced by any character. If number of characters entered are less then the mask size, the field value is completed with zeros (Filling full size field entry is required).
Z	Replaced by any character retrieved from database. Suppress zeros at field left (Complete field filling is optional). When used combined with the mask character X it should be placed at the mask left.

Mask Examples

Mask	Field Value	Formatted Value
(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
(zz) xxxx - xxxx	1234567890	(12) 3456 - 7890
(zz) xxxx - xxxx	12345678	() 1234 - 5678
(zz) xxxx - xxxx	0012345678	() 1234 - 5678

- **Do Not Repeat Value** : Does not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Applies the Regional Settings to format the number of the fields. When not enabled, you will view attributes grouped, Negative Symbol and negative number format.
- **Digit Grouping Separator** : Defines the character that will be used to separate the thousand.
- **Decimal Separator** : Defines the character that will be used to separate the decimal.
- **Negative sign** : Defines the character that will be used to display negative numbers.
- **Negative number format** : Defines the position of the Negative sign em relation to the value.
- **Color of Negative** : Defines a color for the field when the value is negative.
- **Decimal Precision** : Number of decimal places. This value is part of the field size.
- **Complete with Zeros** : Defines the decimal places that will or not be completed zeros to the right. Example: The value is 125,50, but the decimal precision is 3. If the option is enabled, then the value will 125,500.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

Attribute Value Lookup Description

1 (Sports) + 4 (Pleasure) + 8 (Reading) =13

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.

- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.

- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

1. [Home](#)
2. [Applications](#)
3. ReportPDF
4. Percent

Percent

General Settings

Percentage field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Percentage, it is allowed to define the format of a percentage.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Accumulated** : When enables, define that the field will be accumulated with the values of another field. This option is only available on virtual fields.
- **Accumulator field** : Only available when the option "Accumulated" is enabled. This configuration defines which field will be used to accumulate the values. The chosen fields needs to be Integer, Currency or Decimal.
- **Grid Mask** : Defines the mask for the field display. There are two typed of masks described below:

Character	Description
X	Placeholder to any character. Replaced by any character. If number of characters entered are less then the mask size, the field value is completed with zeros (Filling full size field entry is required).
Z	Replaced by any character retrieved from database. Suppress zeros at field left (Complete field filling is optional). When used combined with the mask character X it should be placed at the mask left.

Mask Examples

Mask	Field Value	Formatted Value
(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
(zz) xxxx - xxxx	1234567890	(12) 3456 - 7890
(zz) xxxx - xxxx	12345678	() 1234 - 5678
(zz) xxxx - xxxx	0012345678	() 1234 - 5678

- **Do Not Repeat Value** : Don't repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Applies the Regional Settings to format the number of the fields. When not enabled, you will view attributes grouped, Negative Symbol and negative number format.
- **Digit Grouping Separator** : Defines the character that will be used to separate the thousand.
- **Decimal Separator** : Defines the character that will be used to separate the decimal.
- **Negative sign** : Defines the character that will be used to display negative numbers.
- **Negative number format** : Defines the position of the Negative sign em relation to the value.
- **Color of Negative** : Defines a color for the field when the value is negative.
- **Decimal Precision** : Number of decimal places. This value is part of the field size.
- **Complete with Zeros** : Defines the decimal places that will or not be completed zeros to the right. Example: The value is 125,50, but the decimal precision is 3. If the option is enabled, then the value will 125,500.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1			Sports
2			Culture
4			Pleasure
8			Reading
16			Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1			Sports
2			Culture
4			Pleasure
8			Reading
16			Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute	Value	Lookup	Description
1			Sports
2			Culture
4			Pleasure
8			Reading
16			Music

Attribute Value Lookup Description

1 (Sports) + 4 (Pleasure) + 8 (Reading) =13

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.

- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.

- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

1. [Home](#)
2. [Applications](#)
3. ReportPDF
4. Currency

Currency

General Settings

Currency field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Currency, it is allowed to define the format of a currency.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Accumulated** : When enables, define that the field will be accumulated with the values of another field. This option is only available on virtual fields.
- **Accumulator field** : Only available when the option "Accumulated" is enabled. This configuration defines which field will be used to accumulate the values. The chosen fields needs to be Integer, Currency or Decimal.
- **Grid Mask** : Defines the mask for the field display. There are two typed of masks described below:

Character	Description
X	Placeholder to any character. Replaced by any character. If number of characters entered are less then the mask size, the field value is completed with zeros (Filling full size field entry is required).
Z	Replaced by any character retrieved from database. Suppress zeros at field left (Complete field filling is optional). When used combined with the mask character X it should be placed at the mask left.

Mask Examples

Mask	Field Value	Formatted Value
(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
(zz) xxxx - xxxx	1234567890	(12) 3456 - 7890
(zz) xxxx - xxxx	12345678	() 1234 - 5678
(zz) xxxx - xxxx	0012345678	() 1234 - 5678

- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the datatype of field in the database.

Values Format

Interface of Values Format.

- **Regional Settings** : Allows to apply the Regional Settings to format the number of the fields. When not enabled, you will view attributes grouped, Negative Symbol and negative number format.
- **Currency Format** : Defines the content of the field that if it will be presented with the currency format.
- **Currency Symbol** : Defines the character that represents the Currency Symbol.
- **Digit Grouping Separator** : Defines the character that will be used to separate the thousand.
- **Decimal Separator** : Defines the character that will be used to separate the decimal.
- **Negative sign** : Defines the character that will be used to display negative numbers.
- **Negative number format** : Defines the position of the Negative sign em relation to the value.
- **Color of Negative** : Defines a color for the field when the value is negative.
- **Decimal Precision** : Number of decimal places. This value is part of the field size.
- **Complete with Zeros** : Defines the decimal places that will or not be completed zeros to the right. Example: The value is 125,50, but the decimal precision is 3. If the option is enabled, then the value will 125,500.
- **Display value in full** : The value will be displayed in full. Example: 2018(Two thousand eighteen).

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field

and the value returned from the select command separated by the character in the "Separated by" field.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	

1		Sports	
2		Culture	

Attribute	Value	Lookup	Description
8		Reading	
16		Music	

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- 1. [Home](#)
- 2. [Applications](#)
- 3. ReportPDF
- 4. HTML Image

HTML Image

General Settings

HTML Image field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as HTML Image, it allows to place an image to the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Image** : Defines an image that will be displayed. The icon "Select Image", lists all the standard images from scriptcase and also the image that you have uploaded to scriptcase. The icon "Upload an image" allows the developer to send an image to the server which is from another machine.
- **Border** : Border size for the image in Pixels.
- **Width** : Define the width of the image in Pixels.
- **Height** : Define the height of the image in Pixels.

- 1. [Home](#)
- 2. [Applications](#)
- 3. ReportPDF
- 4. Credit Card

Credit Card

General Settings

Credit Card Number Configuration Interface.

- **Data Type** : Select the type of field for the application. When it is defined as a Credit Card Number, you can define some rules for the display format of the Credit Card.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name would be "Client Name".
- **Do Not Repeat Value** : Do not repeat the value of the field in the case it is the same as the previous record.

- **SQL Type** : Informs the data type of the field in the database.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced by the curly brackets {}. While running the application, the field within the curly brackets is replaced by its value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label Value Start Size

Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- 1. [Home](#)
- 2. [Applications](#)
- 3. ReportPDF
- 4. E-mail

E-mail

General Settings

Email field Configuration Interface.

- **Data Type** : Defines the type of field for the application. When it is defined as an Email, when you click on the field you be offered a choice for your email client and send an email to that specific email.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
- **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
- **Summary** : Applies the lookup only for the summary (HTML and PDF).
- **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value**: When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Martial Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

1. [Home](#)
2. [Applications](#)
3. ReportPDF
4. URL

URL

General Settings

URL field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a URL, it sets the value as a clickable link (Only if it contains a valid path).
 - **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
 - **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.
-
- **SQL Type** : Informs the data type of field in the database.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

• Lookup Methods

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

• Lookup Method - Automatic

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced by the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Martial Status, the Hobby will use two bytes.

The example below:

Label Value Start Size

Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- 1. [Home](#)
- 2. [Applications](#)
- 3. [ReportPDF](#)
- 4. [Date](#)

Date

General Settings

Date field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Date and Time, you can define the format of the date.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

Format of Values without Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator attribute (Check the Regional Settings).
- **Display** : Offers a series of formats predefined for displaying dates.
- **Detail Mask** : Define the format of the field in the Grid, following the standards the PHP function `Date`.

d-m-Y	25-09-2009
F/Y	September/2009
j/n/Y \a\s g:i:s A	25/9/2001 as 14:30:11 PM
l, d \d\l e F \d\l e Y	Thursday, 25 of January of 2009
h:i:s	11:33:20
#h:i:s	123:43:27 (accumulating the hours)

- **Format Table**

- **Internal Format** : Allows to define the format the field when the SQL type is different from **Date**. You need to use the characters **Y**, **M** and **D** that correspond to **Year**, **Month** and **Day**.
 - **Example 1** : Date stored in a SQL field of the type **char(8)** where the positions **1** to **4** represents the year, the positions **5** and **6** is the month and the position **7** and **8** is the day, define the internal format like: **YYYYMMDD**
 - **Example 2** : Date stored in a SQL field of the type **char(6)** where the positions **1** and **2** represents the month and the positions **3** to **6** is the year, define the internal format like: **MMYYYY**

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets `{}`. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then `(;)`.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does

not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Martial Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1			Sports
2			Culture
4			Pleasure
8			Reading
16			Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1			Sports
2			Culture
4			Pleasure
8			Reading
16			Music

1			Sports
2			Culture
4			Pleasure
8			Reading
16			Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute	Value	Lookup	Description
1			Sports
2			Culture
4			Pleasure
8			Reading
16			Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.

- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.

- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- 1. [Home](#)
- 2. [Applications](#)
- 3. [ReportPDF](#)
- 4. [Time](#)

Time

General Settings

Time field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as Time, it is possible to define a Time Format.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

Format of Values without Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator attribute (Check the Regional Settings).
- **Display** : Offers a series of formats predefined for displaying dates.
- **Detail Mask** : Define the format of the field in the Grid, following the standards the PHP function Date.

```
d-m-Y      25-09-2009
F/Y       September/2009
j/n/Y \a\s g:i:s A 25/9/2001 as 14:30:11 PM
l, d \d\e F \d\e Y Thursday, 25 of January of 2009
h:i:s     11:33:20
#h:i:s    123:43:27 (accumulating the
           hours)
```

- **Format Table**

- **Internal Format** : Allows to define the format the field when the SQL type is different from **Date**. You need to use the characters **Y**, **M** and **D** that correspond to **Year**, **Month** and **Day**.
 - **Example 1** : Date stored in a SQL field of the type **char(8)** where the positions **1** to **4** represents the year, the positions **5** and **6** is the month and the position **7** and **8** is the day, define the internal format like: **YYYYMMDD**
 - **Example 2** : Date stored in a SQL field of the type **char(6)** where the positions **1** and **2** represents the month and the positions **3** to **6** is the year, define the internal format like: **MMYYYY**

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets **{}**. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does

not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Martial Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.

- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.

- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- 1. [Home](#)
- 2. [Applications](#)
- 3. ReportPDF
- 4. Date and time

Date and time

General Settings

Date and Time field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Time, you can define the format of the time.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

Format of Values without Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator attribute (Check the Regional Settings).
- **Display** : Offers a series of formats predefined for displaying dates.
- **Detail Mask** : Define the format of the field in the Grid, following the standards the PHP function Date.

d-m-Y	25-09-2009
F/Y	September/2009
j/n/Y \a\s g:i:s A	25/9/2001 as 14:30:11 PM
l, d \d\le F \d\le Y	Thursday, 25 of January of 2009
h:i:s	11:33:20
#h:i:s	123:43:27 (accumulating the hours)

- **Format Table**

- **Internal Format** : Allows to define the format the field when the SQL type is different from **Date**. You need to use the characters **Y**, **M** and **D** that correspond to **Year**, **Month** and **Day**.
 - **Example 1** : Date stored in a SQL field of the type **char(8)** where the positions **1** to **4** represents the year, the positions **5** and **6** is the month and the position **7** and **8** is the day, define the internal format like: **YYYYMMDD**
 - **Example 2** : Date stored in a SQL field of the type **char(6)** where the positions **1** and **2** represents the month and the positions **3** to **6** is the year, define the internal format like: **MMYYYY**

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets `{}`. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then `(;)`.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does

not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Martial Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1			Sports
2			Culture
4			Pleasure
8			Reading
16			Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1			Sports
2			Culture
4			Pleasure
8			Reading
16			Music

1			Sports
2			Culture
4			Pleasure
8			Reading
16			Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute	Value	Lookup	Description
1			Sports
2			Culture
4			Pleasure
8			Reading
16			Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.

- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.

- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- 1. [Home](#)
- 2. [Applications](#)
- 3. ReportPDF
- 4. Image (Database)

Image (Database)

General Settings

Upload field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as an Image (Database), all the Images files are stored and loaded directly from the Database.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Image Border** : Width of the Image border. Value in Pixels.
- **Image Height** : Image height size. Value in Pixels.
- **Image Width** : Image width size. Value in Pixels.
- **Maintain Aspect** : Maintains the original aspect ratio of the image when resizing it.
- **Open in Another Window** : Allows to open the image in another window.
- **SQL Type** : Informs the data type of field in the database.

- 1. [Home](#)
- 2. [Applications](#)
- 3. ReportPDF
- 4. Image (File Name)

Image (File Name)

General Settings

Upload field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as an Image (File Name), all the images files are stored and loaded in a directory of the server (Only the image name is stored in the database).
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Image Border** : Width of the Image border. Value in Pixels.
- **Image Height** : Image height size. Value in Pixels.
- **Image Width** : Image width size. Value in Pixels.
- **Maintain Aspect** : Maintains the original aspect ratio of the image when resizing it.
- **Open in Another Window** : Allows to open the image in another window.
- **SQL Type** : Informs the data type of field in the database.

- 1. [Home](#)
- 2. [Applications](#)
- 3. ReportPDF
- 4. Document (Database)

Document (Database)

General Settings

Upload field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Document (Database), all the document files are stored and loaded directly from the Database.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Icon** : Displays an icon beside the field to identify the type of document.
- **File Name** : Defines the field that contains the name of the document stored in the database (the field that contains this information also needs to be stored in the database).
- **SQL Type** : Informs the data type of field in the database.

1. [Home](#)
2. [Applications](#)
3. ReportPDF
4. Document (File Name)

Document (File Name)

General Settings


Upload field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Document (File Name), all the document files are stored and loaded in a directory of the server (Only the document name is stored in the database).
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Sub-folder** : Sub-folder name that the files are stored.
- **Icon** : Displays an icon beside the field to identify the type of document.
- **File Name** : Defines the field that contains the name of the document stored in the server (the field that contains this information also needs to be stored in the database).
- **SQL Type** : Informs the data type of field in the database.

- 1. [Home](#)
- 2. [Applications](#)
- 3. ReportPDF
- 4. QRCode

QRCode

General Settings

 QRCODE field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a QRCODE, allows you to set values into a QRCODE.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **SQL Type** : Informs the data type of field in the database.

Watch below a video showing an example with QRCODE



Values Format


 Interface of Values Format.

- **Level of error correction**: Codewords are 8 bits long and use the Reed-Solomon error correction algorithm with four error correction levels. The higher the error correction level, the less storage capacity.
- **Image Size** : Size of the QRCODE.
- **Margin** : Margin of the QRCODE.

 Interface of Values Format.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

 Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. ReportPDF
- 4. Bar code

Bar code

General Settings

 Configuration Interface of the Barcode Field.

- **Data Type** : DataType of the field for the application.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name would be "Client Name".
- **SQL Type** : Database field type.

Watch below a video showing an example about the Barcode field:



Values Format

 Configuration Interface of the Barcode Field.

- **Type** : Type of Barcode.
- **Text** : Barcode Text for illustration purposes.
- There are **18 types of barcodes** that are listed below:

 Barcode configuration interface.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

 Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.



1. [Home](#)
2. [Applications](#)
3. ReportPDF
4. SubSelect

SubSelect

General Settings

SubSelect field configuration Interface.

Increment : Defines the space between the record lines of the SubSelect field. **Connection** : Displays a list of connections created in the current project. **Table** : Displays the tables related to the database connection selected previously. **SubSelect** : Displays the SQL of the chosen table in the previous option. Allowed to be modified and use field variables and global variables. **SQLBuilder** : Opens the ScriptCase SQLBuilder, to build the SQL commands.

Scriptcase uses the events to enable the developer to customize the application code. Using the events you can program custom actions at a specific execution time (e.g. After a record is inserted, when it is loaded, upon submitting a form, ...) and for a specific application type. In the event areas you can use global and local variables, JavaScript, CSS, HTML, PHP codes and also Scriptcase macros.

onApplicationInit

This event occurs only once, when the application is loaded and before the application executes the SQL. It can be used for data processing or checking variables.

Example: `if ([glo_var_dept] != 'financial'){ sc_redir(app_x.php); }`

onNavigate

This event occurs when navigating between the application pages.

OnScriptInit

This event occurs when the application scripts are started. It runs before the OnLoad events(when present), however, it will always be executed when running, reloading or navigating the application.

onRecord

This event is executed at the moment the record line is loading. The OnRecord event is a loop that loads the records by each row.

onHeader

This event runs when the header of the application is loaded.

onFooter

This event runs when the footer of the application is loaded.

Settings

With this interface, you can define general options of the Search Form.

ATTRIBUTE	VALUE
Search Criteria	AND ▼
Display Condition	<input type="checkbox"/>
Use auto-complete in the fields.	Defined in the field ▼

Search configuration Interface.

Search Criteria

Allows to select the logical operator **AND** or **OR** to define the criteria of the search;

Display Condition

Gets the condition of the search available for the user to choose one. He can select "AND" or "OR" in a Combobox.

Use auto-complete in the fields

Automatically turns the field into an autocomplete according to the existing values in the database. If the user chooses **Yes**, the autocomplete will enable automatically in all inputs that contain a relationship. If the user decides **No**, so no autocompletes will be displayed. Otherwise, the option selected is **Defined in the field** it'll keep the settings for each field individually.

Search Criteria

With this interface, you can configure the conditions available for each field of the Search form.

ATTRIBUTE	VALUE	DESCRIPTION
Check the field that is part of the search and select in the right combo-box an option to enable or disable a search option.		
customerid	*Contains	
companyname	*Not Contains	
contactname	*Equal to	
contacttitle	*Empty	
birthdate	Different	
country	Beginning with	
regionid	Greater than	
stateid	Greater equal	
city	Less than	
address	Less equal	
postalcode	Between two values	
phone	.in	

On/Off
All
None

Search configuration Interface.

We can see the fields list on the left combo. On the right, the list of options for filtering the selected field. To select an option, click on one of them (Equal to, Beginning with, Contains, etc.) and then the button On/Off. The arrows, on the right, allows altering the order of the fields.

For the Date type fields, you can define special conditions for the search, accessing the field configurations, and editing the Special Conditions Settings.

Below the list are the buttons to enable the selected options:

- **On/Off:** Enables or disables the field or the option chosen.
- **All:** Marks all fields or options.
- **None:** Unmarks all the fields or options.

Advanced Search

Settings

Through the table below we are able to set all the options that will be part of the application "Grid Search".

Margins

Defines the position of the margins of the Search Form.

Keep Values

It keeps the searched values when the user returns to the search form.

Keep Columns and Order Selection

Set it to preserve the selected columns and sorting for each search, if they went changed by the user through the toolbar options.

Use Enter to

It allows you to define the action that the Enter Key has on the Search form. **Tabulate** enables you to navigate between fields, and **Submit** performs the search(activates the Search button).

Display Tags

Allows displaying as tags, the searches used for the Grid.

Display after filtering

Display tags only after performing an advanced search. If disabled, it will always display a tag, regardless of the advanced search.

Unify results

Sets the chars limit to group the result of the tags. This option should be used when the field type is multiple-select.

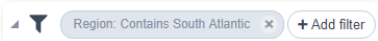
Treeview in the Tags

Sets the use of Treeview for tags.

Initial status of the Treeview

It sets the initial state of the Treeview. (If the app is using Treeview for tags)

Start open

It displays the tags. 

Start close

It displays the full description of the tags as text. 

Select Fields

![[Interface for filter fields selection.][filtro_avancado_configuracao_selecionar_campos] Interface for filter fields selection.

Required

Defines which fields of application will be required for the Search.

![[Required fields interface.][filtro_avancado_configuracao_campos_obrigatorios] Required fields interface.

The application generated will be displayed a bullet () next to the field and an error message is generated if not assigned no value.*

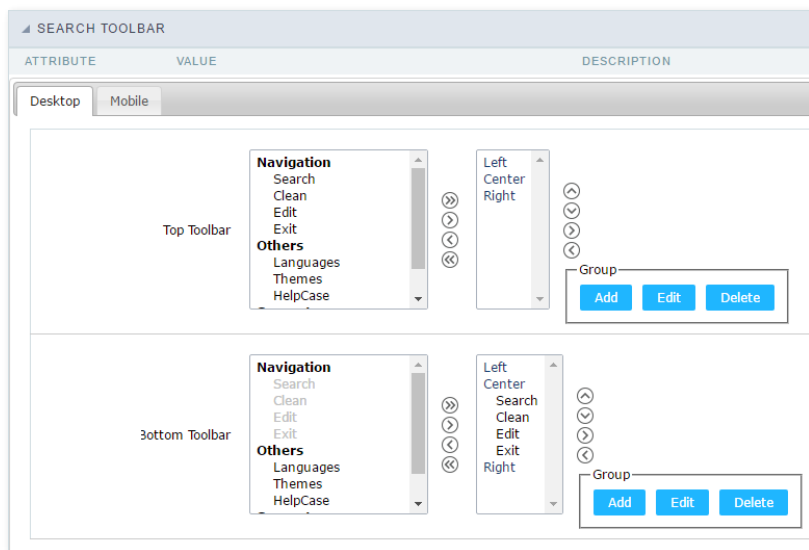
![[Configuration interface of the marker placement.][filtro_avancado_configuracao_campos_obrigatorios_posicionamento] Configuration interface of the marker placement.

- **Marker position** : Marker's position relative to the field.
- **Display message** : Displays whether or not the validation error message.

Toolbar

Desktop

The Search toolbar is divided in two parts: Top and Bottom, in a way that is possible to define the buttons that will be displayed in both bars. The selection of buttons in the top and bottom toolbar works independently, allowing the buttons to be displayed in both bars at the same time.



Toolbar Interface.

Navigation: Groups the options relative to the navigation buttons that can be displayed in the application.

- **Search:** Execute the search.
- **Clean:** Clean the all the search fields.
- **Edit:** Enable the *Save Tag* option.

- **Exit:** Exit the application.

Others: Groups a diversity of options relative to the application.

- **Languages:** Displays a combobox with the names available, defined in the project properties.
- **Themes:** Displays a combobox with the themes available, defined in the project properties.
- **HelpCase:** Displays a button to redirect to the help page.

Separator

- : Displays a line separating the buttons, when used the Group Buttons.

Use in-line buttons: Allows the alignment of the filter buttons next to the fields.

- **__ Inline buttons__:** Allows you to select which buttons will be displayed next to the field, and you can sort them according to your wishes. This option is available by enabling **Use Online Buttons** in the button settings.

Button Settings

Button	Label	Hint	Shortcut key
Search			
Clean			
Edit			
Exit			

Button Settings Interface.

- **Hotkey:** Allows you to set keyboard hotkeys to a button.
- **Use Inline Buttons:** Allows alignment of the filter buttons next to the fields.
- **Position of the in-line buttons:** Sets the positioning of the buttons to the right or left of the fields.
- **Column Quantity:** Sets the number of display columns of the buttons, allowing you to configure whether they will be displayed side-by-side or distributed in columns.

Options

ATTRIBUTE	VALUE
Buttons Position (top)	Center ▾
Buttons Position (bottom)	Center ▾

Options Interface.

- **Button Position(Top/Bottom)** Positioning the buttons of the toolbar Top/Bottom.

Save Search

This feature allows the end-user to save his searches in a profile. You can create some rules, like to save the searches by user login.

SAVE CRITERIA	
<input checked="" type="checkbox"/> Enable Public	
Title	(lang_srch_public)
<input checked="" type="checkbox"/> Use Rules	
	<div style="border: 1px solid red; padding: 2px; display: inline-block;">New rule</div> <div style="margin-left: 5px;"> <div style="border: 1px solid blue; padding: 2px; display: inline-block;">Edit</div> <div style="margin-left: 5px; border: 1px solid blue; padding: 2px; display: inline-block;">Delete</div> </div>

Save Filter Interface.

SAVE CRITERIA	
Name	[var_login] ▾
Variable name	
Label	
	<div style="border: 1px solid red; padding: 2px; display: inline-block;">Add</div> <div style="margin-left: 5px; border: 1px solid blue; padding: 2px; display: inline-block;">Delete</div>
Finish	Cancel

Save Filter Interface.

Events

In event blocks can be used global variables, local, JavaScript code, CSS codes and Scriptcase macros.

onScriptInit

This event occurs when the application scripts are started. It runs before the OnLoad events(when present), however, it will always be executed when running, reloading or navigating the application.

onRefresh

This event runs when the refresh option of the application is enabled.

onSave

This event runs when the application saves the record.

onValidate

This event runs when validating the information of the fields, when submitting the form.

Layout

This module lets you editing display options for the application. You can apply visualization themes, organize blocks, define values, and the display format of the Header and Footer.

Blocks

Blocks are “containers” where you can position the application fields of Forms, Controls, or Slide Grids.

Scriptcase creates applications with one block by default. You can add more blocks as you wish, to organize it in the best way.

See below, the Columns Organization, and where you can define the position of the next block: beside or below the current one.

Block		Title		Label		Fields		Organization		
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
	form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Pag2										
	Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Blocks not shown										

Application Block configuration

On the left side of each block, there are two icons, the first one to edit the information of the block and the second one to delete the block.

Organizing The Position Of The Blocks

See below how to modify the display order of the Blocks in one Page.

Click and drag the block that you desire to modify to its new position.

Block		Title		Label		Fields		Organization		
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
	form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Pag2										
	Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Blocks not shown										

Application Block Display configuration

- See how to remove a block from the display

Click on the block desired and drag it to the item “Blocks not Shown”. This way, you can also drag the block to another page if desired. See the images below.

Block		Title		Label		Fields		Organization		
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
	form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Pag2										
	Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Blocks not shown										
	Block 3	Block 3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼

Application Block Display configuration

Block		Title		Label		Fields		Organization		
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
	form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Pag2										
	Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Blocks not shown										
	Block 3	Block 3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼

Application Block Display configuration

Block

- **Name:** The name that identifier the Block.
- **Label:** Title of the block to display in the application.

Title

- **Display:** It controls the display of the block title.

Label

- **Display:** It controls the display of the field labels of the block.
- **Position:** Options to display label :

1. **Above:** Display the label above the field.
2. **Beside:** Display the label beside the field.
3. **Below:** Display the label below the field.

Fields

- **Columns:** Amount of columns side by side in the block.
- **Position:** The way to display the fields in the block:
 1. **Below:** Display the fields one below the other respecting the number of columns.
 2. **Beside:** Display the fields one beside the other respecting the number of columns.
 3. **Line:** Display the fields one beside the other with no tabulation.

Organization

- **Next:** The way to display the blocks in the page:
 1. **Below:** Set to show the following block below the current one.
 2. **Beside:** Set to show the following block beside the current one.
 3. **Tabs:** Set to show the following block in a different tab then the current one.
- **Width:** Set the block width in pixels or percentages. Use the symbol "%" to indicates the value in percentage.
- **Collapse:** Enables the option to close the block.

Create A New Block

To include new blocks in an Application, click on the button **Create New Block**. Then, enter the name and label of the block in the following interface and finish by click on Create.

Creating application blocks configuration


Name

Name of the Block.

Label

Title of the block to display in the application.

Edit Blocks

To edit a block, click on the icon  that is on the left side of the block. Then you can see the following interface to define the parameters of the blocks. Click on Save to finish.

ATTRIBUTE	VALUE
Name	form_orders
Title	form_orders
Display Title	<input type="radio"/> Yes <input checked="" type="radio"/> No
Title Font	<input type="text"/> Aa
Font Size	<input type="text"/>
Font Color	<input type="text"/>
Background Color	<input type="text"/>
Background image	<input type="text"/>
Title Height	<input type="text"/> 20 pixels
Horizontal Alignment	<input type="text"/>
Vertical Alignment	<input type="text"/>
Display Label	<input checked="" type="radio"/> Yes <input type="radio"/> No
Columns	<input type="text"/> 1
Columns Width	Calculated
Label Color	<input type="text"/>
Fields Organization	Beside
Label Position	Beside
Next Block	Below
Border Color	<input type="text"/>
Border Width	<input type="text"/> 0 pixels
Block Width	100%
Block Height	<input type="text"/>
Cell Spacing	<input type="text"/> pixels
Collapse	Start open

Application Block editing interface

Name

Name of the block. ##### Title
 Block title for display. ##### Display Title
 This option, when active, allows displaying the block title. ##### Title Font
 Set the font family of the block title. ##### Font Size
 Set the font size of the block title. ##### Font Color
 Set the font color of the block title. ##### Background Color
 Set the Background Color of the block title. ##### Background image
 Set a Background image for the block title. ##### Title Height
 Height in pixels of the block title line. ##### Horizontal Alignment
 Horizontal Alignment of the block title (Left, Center, and Right). ##### Vertical Alignment
 Vertical Alignment of the block title (Top, Middle, and Bottom). ##### Display Label
 Display the labels of the fields in the block. ##### Columns
 Amount of field columns in a block. ##### Columns Width
 Set the field column width of the block. ##### Label Color
 Color of the field labels. ##### Fields Organization
 The way to display the fields in the block. ##### Label Position
 Set the position of the field labels of the block. ##### Next Block
 Set the position of the following block relating to the current one. ##### Border Color
 The border Color for the block. ##### Border Width
 The border Width for the block. ##### Block Width
 The width for the block. ##### Block Height
 The Height for the block. ##### Cell Spacing
 The Cell Spacing in the block. ##### Collapse
 It enables the option to close the block.

Layout Settings

ScriptCase creates an app by using default values defined per project, but you can change those values for a specific application on this interface.

LAYOUT SETTINGS

Header Template	Flat	Template name used for the application header.
Footer Template	Default	Template name used for the application footer.
Button		Use different buttons than what was defined in the color scheme.
Themes	Sc9_Rhino	Use different themes from the one defined for the color scheme

Header

Navigation: |< < > >| xxyyzz xxxxx yyyy

Block 1

Name: xxxxxxxxxxxx

Type: Male Female

Address*: xxxxxxxxxxxx


Groups*: Male Female

Countries: Afghanistan

Address: yyyyyyyyyyyyyyy

Photos: Drag & Drop files here

Image1.png ✓
Image2.png ✗

Captcha: 

Application Layout Settings

Header Template

Allows choosing the look of the application Header.

Footer Template

Allows choosing the look of the application Footer.

Button

Allows choosing the button theme for the application.

Themes

Choose one of the selected themes in the project properties. It defines the look of the application, like colors, fonts, and others.

Header & Footer

Header

Here we can define if we want to display the header or not, the application title, and fill the header variables. The header looks according to the selected template in the Layout settings interface.

HEADER	
ATTRIBUTE	VALUE
Display Header	<input checked="" type="checkbox"/>
Insertion Title	{lang_othr_frmi_tit} - orders
Update Title	{lang_othr_frmu_tit} - orders
Header Variables	
SC_HEADER	Title ▾
SC_VALUE	Date ▾ <input type="text"/> ?

Application Header Configuration

Display Header


This option determines if we want to display the header or not.

Title

Allows informing the title of the application.

Header Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **Field:** When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **Title:** It displays the value of "**Application Title**" in the header.
- **Date:** It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **Image:** It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **Value:** It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Depending on the Application, you may have more than one title option.

Footer

FOOTER	
ATTRIBUTE	VALUE
Display Footer	<input type="checkbox"/>
Footer Variables	
NM_FOOTER_DATA	▾
NM_FOOTER_LOGO	▾
NM_FOOTER_TITLE	▾

Application Footer Settings


This page may change depending on the footer template chosen in the Layout Settings.

Display Footer

This option determines if we want to display the footer or not.

Footer Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **Field:** When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **Date:** It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **Image:** It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **Value:** It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Fields

This is a list of fields available for use in the filter. Click on the desired field to access the configuration instructions.

Text Field	Field Select
Integer Field	Field Double Select
Decimal Field	Field Checkbox
Currency Field	Field Radio
Date Field	Field Text Auto-Complete
Field time	Field Number Auto-Complete
Field Datetime	

Settings

With this interface, you can set the common attributes of the app.

ATTRIBUTE	VALUE
Application Code	grid_customers (9.00.0000)
Description	<input type="text"/>
Documents Path	C:/Program Files/NetMake/v9/www/
Image Directory	/scriptcase/file/img
Application images	<input type="button" value="Add"/> <input type="button" value="Remove"/>
Language	English (United States) ▾
Share Location Variable	<input checked="" type="checkbox"/>
Charset	<input type="text"/>
Share Theme Variable	<input checked="" type="checkbox"/>
Folder	root ▾
Edit by Project	<input checked="" type="checkbox"/>
Timeout	<input type="text" value="0"/>
HelpCase Link	Application ▾ Search ▾ Summary ▾

Application Settings Interface

- Attributes**
 - Application Code** : It is the name that defines an application. An app can be renamed at the [List of Application](#).
 - Description** : This field contains a brief description of the application objectives.
 - Documents Path** : The absolute path to store uploaded documents in the application.
 - Image Directory** : The filesystem directory to store the application images.
 - Application images** : Import images into the application to allow using them in the application.
 - Language** : Set the default language of the application. Display all the application hints and messages in the selected language.
 - Share Location Variable** : Define if the app shares the regional settings with other applications through a session variable.
 - Charset** : Define a specific charset to use in the application.
 - Share Theme Variable** : Define if the app shares the Theme settings with other applications through a session variable.
 - Folder** : Define the project folder that contains the app.
 - Edit by Project** : Define if other project developers can edit the application.
 - Timeout** : Set the session runtime timeout in seconds. If the value is Zero, it assumes the default timeout of the PHP.
 - HelpCase Link** : It allows to associate a [HelpCase](#) file with the application.

Notification Settings

ATTRIBUTE	VALUE
Use SweetAlert	<input type="checkbox"/>
Error Position on the field	Down ▾
Show the Error Title in the Application	<input checked="" type="checkbox"/>
Show the Error Title in the Field	<input type="checkbox"/>
Error Title	{lang_errm_errt}
Script Error	<input type="checkbox"/>
SQL Error	<input checked="" type="checkbox"/>
Debug Mode	<input type="checkbox"/>
Ajax Error Output	<input checked="" type="checkbox"/>

- Use SweetAlert**: Use the SweetAlert to display messages from the application. When this option is active, it will replace the browser's "confirm" and "alert".
- Use SweetAlert position using Toast** : The position to display error messages on the application.

- **error Position on the field** : The position to display error messages when criticizing the field.
- **how the Error Title in the Application** : Define to display the title line of the error message or not.
- **how the Error Title in the Field** : Define to display the title line of the error message in the field or not.
- **cript Error** : Allows displaying the line code where there is an error..
- **QL Error** : Allows displaying the SQL statement if it got an error.
- **ebug Mode** : Runs the application in Debug mode, showing all SQL statements the application is executing.
- **jax Error Output** : Enables the Ajax alert for debugging errors.

Navigation

This interface allows defining the navigating behavior of the application

NAVIGATION	
ATTRIBUTE	VALUE
Exit URL	<input type="text"/> 🔗
Close on Exit	<input type="checkbox"/>
Redirect URL	<input type="text"/> 🔗
Redirect Variable	<input type="text"/>

Navigation Interface.

Exit URL

URL to where the user goes when he clicks on the “exit” button.

Close on Exit

Close the browser window when the user clicks on the “exit” button.

Redirect URL

Redirect to another URL in case there aren't any global variables available.

Redirect Variable

Creates a variable with the application name and sends it to the redirected application.

Global Variables

This screen shows all global variables used in the application

The global variable is an external parameter required for running the application. The application can include global variables in the WHERE clause, field definitions and names, event programming, and so on.

You can define global variables by using square brackets ([variable]). You can pass the parameters to the application through one of the methods: Session, Post, and Get

if you need to use [Database Identifiers](#) in or tables, we recommend using the double quotation marks(" ") instead of the brackets([]) because of the conflict with the [Global Variables syntax](#) of Scriptcase. Using the brackets as database identifiers may lead to problems in the Scriptcase applications.

APPLICATION	
Settings	
Navigation	
Messages	
Global Variable	

Global variables Interface.

Example:

```
Select CustomerID, CustomerName from Customers WHERE CustomerID = '[v_customerid]'
```

We do not recommend using session variables (global variables) with the same name as the table fields.

In this case, the variable **v_customerid** is displaying in the global variables configuration.

VARIABLE SETTINGS	
ATTRIBUTE	VALUE
global	<div style="border: 1px solid black; padding: 5px;"> <p>Scope</p> <p><input type="checkbox"/> SESSION</p> <p><input checked="" type="checkbox"/> POST</p> <p><input checked="" type="checkbox"/> GET</p> </div> <div style="border: 1px solid black; padding: 5px; margin-top: 5px;"> <p>Settings</p> <p><input type="checkbox"/> Optional</p> </div> <div style="border: 1px solid black; padding: 5px; margin-top: 5px;"> <p>Type</p> <p><input type="radio"/> Out</p> <p><input checked="" type="radio"/> In</p> </div>

Global variables configuration Interface.

Attribute

It shows the variable's name in the application.

Value

Defines the behavior of the variables, divided into three blocks, they are:

Scope

Defines how the application receives the variable. A variable defined as the POST method in the app does not receive value if it comes with the GET method. Those are the methods available:

- **ession**: Defines that the variable must be created as a PHP session variable by another application.
- **ET**: Defines that the variable must be passed by the URL, that is, visible by the browser.
- **OST**: Defines that the variable must be passed through the POST method of an HTML form.

Settings

- **ptional**: Check it to don't validate the variable at runtime.

Type (In / Out)

Defines whether the variable comes from another application (In), or it is starting in the current app (Out).

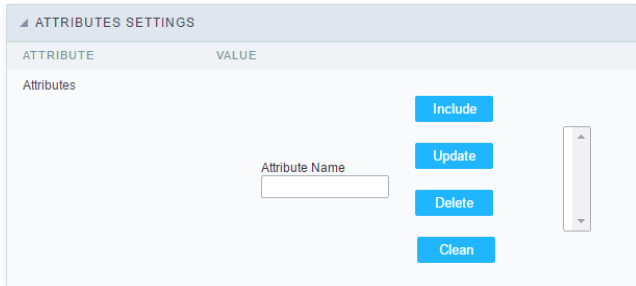
Description

Indicates where the application is using the variable.

In this version of ScriptCase is incorporated the concept of programming with the use of attributes, methods, resources and libraries. In the previous version it was already possible to create business rules in applications using this concept. The big difference now is that this can be done in a more organized, facilitating both the development as the understanding of the rule by another developer.

Attributes

The attributes are variables that has the global scope declared in the application. An attribute that can be referenced in all the methods and the application's events.



Attributes configuration Interface.

- **attributes**
 - **attribute Name** : This field allows to inform the attributes name, after typing the name just click on **include**. In case you want to **update** click on the attributes name (located on the buttons' right side), do the necessary modifications and click on update. In case it is necessary to **delete** an attribute, it is necessary to do same process as the update, but when selecting the attribute, it is necessary to click on the delete button. The **clear** button, as the name suggests, it clears the list of all the attributes listed on the right.

Internal Libraries

To use the internal libraries just select the correspondent (Project Libraries for example). This process allows the methods existing in the library, to be visible in all the application's Events and Methods. See how to manage the libraries by [clicking here](#).

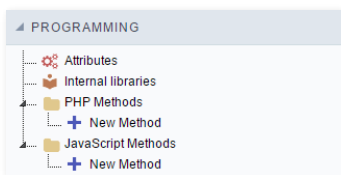


Internal Libraries management Interface

PHP Methods

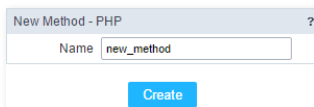
Methods are function or procedures declared by the developer, that helps when applying the routines. Using methods in the applications allows to reuse your code throughout the application, optimizing the development experience.

Creating A New Method



- Give the method a name and click Create. As in the image below.

PHP methods are incorporated into the same class, so it is not possible to use two functions with the same name even when used in different methods.



- Methods can receive parameters.

```
function new_method
1 echo "Hello World!!!";
```

- dd the amount of variables:

Definition of the parameters of the method:

new_method

- No defined parameter.

Add Parameter(s) Cancel

Save

- efining the variables:

Insertion of Parameters

Name	Type	Value Standard
<input type="text"/>	For Value	<input type="text"/>
	For Value	
	For References	

Save Back Cancel

- **ame** : Type in the variable's name.
- **ype** : Selecting the type of variables: For Value or For Reference.
- **alue Standard** : The parameter's value used to initialize when calling the method.

References allows to create a second name for a variable that you can use to read and modify the variable's original information.

- diting a parameter:

Definition of the parameters of the method:

new_method

Parameters

Stest = test

⏪ ⏩

☑ ☐ ✎ ✕

Add Parameter(s) Cancel

Save

- ☑ : Checking all the parameters in the parameter list.
- ☐ : Uncheck all the parameters in the parameter list.
- ✎ : Edit the selected parameter of the list.
- ✕ : Deletes the selected variable of the list.

The PDF Report application has a way of positioning the report fields in a simple and dynamic way, using the drag and drop functionality.

In the PDF Layout, at the main screen, you can change some application settings. These settings are also present in PDF Report > Settings.



Using Drag and Drop functionality, you can configure the placement of fields in this interface. To change them you need to click on the field you prefer to change the position and drag to their new position.

Clicking on a specific field opens a field configuration screen.

Properties

Properties
General Settings
Lookup settings

Name	<input type="text" value="cell_companynome"/>	?
Field	<input style="border: none; background-color: #f0f0f0; width: 100%;" type="text" value="companynome"/>	?
Pos X	<input type="text" value="10"/>	?
Pos Y	<input type="text" value="20"/>	?
Width	<input type="text" value="0"/>	?
Alignment	<input type="text" value="Left"/>	?
Font	<input type="text" value="Arial"/>	?
Font Size	<input type="text" value="0"/>	?
Font Color	<input style="width: 20px; height: 15px; background-color: black; border: none;" type="color"/>	?
Font Style	<input type="checkbox"/> Bold <input type="checkbox"/> Italic <input type="checkbox"/> Underline	?
Show	<input checked="" type="checkbox"/>	?

- **ame:** On this field, it is necessary to inform a name to identify the field.
- **ield:** On this option, you need to select the field from the table that will be used in the cell. In case you have selected a field from the "Special" category you will see the following:
 - **ystem Date:** The current date of the system on which the application runs.
 - **urrent Page:** Number of the current page that the cell is located.
 - **otal Page:** Total number of application pages.
 - **abel:** Selecting this option will open an input so you can inform the text that will be displayed.
- **os X:** Positioning the X axis. Up and down.
- **os Y:** Positioning the Y axis. Left and Right.
- **idth:** Enter the width of the cell. The unit of value is pixel.
- **ignment:** In this field, you will select the alignment of the cell text: Left, Right or Center.
- **ont:** In this field, you will select a source according to the list.
- **ont Size:** Here you inform the font size of the text that will be displayed.

- **ont Color**: In this field, you will inform a color in hexadecimal. (Example: # 000000)
- **ont Style**: In this field, you will mark the style of the font: Bold, Italic and / or Underline. (You can select more than one option)
- **how**: In this field, you enable or disable the field display.

General Setting - Text Fields

- **ata Type** : Define the type of field for the application.
- **ase Settings** : Convert the letter from the field when losing focus. The options are:
 - **pper case** : All in Upper Case.
 - **ower case** : All in Lower case.
 - **apitalize first word** : Capitalizes the first letter of the first word.
 - **apitalize all words** : Capitalizes the first letter of all the the words.
- **how HTML content** : Determines if the HTML contained in the field will be displayed or not. If enabled, the html will be displayed, otherwise the html will be interpreted by the browser.
- **mount of characters**: Defines the field width. This option is available only for the field type **Multiple lines text**.
- **ield Mask**: In this field, you can configure a display mask according to the table given in Applications> Grid > Fields> Text.
- **QL Type** : Informs the data type of field in the database.

General Setting - Integer Fields

- **ata Type** : Define the type of field for the application.
- **ield Mask**: In this field, you can configure a display mask according to the table given in Applications> Grid > Fields> Text.
- **QL Type** : Informs the data type of field in the database.
- **egional Settings** : Allows to apply the Regional Settings to format of the fields. When not enabled, you will view attributes for set yourself. You can change of the Regional Settings in **Locales** >**Regional settings**. For more information [click here](#).
- **olor of Negative**: In this field, you will inform a color in hexadecimal. (Example: # 000000)

General Setting - Currency Fields

- **ata Type** : Define the type of field for the application.
- **ield Mask**: In this field, you can configure a display mask according to the table given in Applications> Grid > Fields> Text.

- **QL Type** : Informs the data type of field in the database.
- **Regional Settings** : Allows to apply the Regional Settings to format of the fields. When not enabled, you will view attributes for set yourself. You can change of the Regional Settings in **Locales** > **Regional settings**. For more information [click here](#).
- **urrency Format** : Defines the content of the field that if it will be presented with the currency format.
- **olor of Negative**: In this field, you will inform a color in hexadecimal. (Example: # 000000)
- **ecimal Precision** : Number of decimal places. This value is part of the field size.
- **omplete with Zeros** : Defines the decimal places that will or not be completed zeros to the right. Example: The value is 125,50, but the decimal precision is 3. If the option is enabled, then the value will 125,500.

General Setting - Decimal Fields

Properties | General Settings | Lookup settings

Data Type: Decimal

Field Mask:

Repeat value:

SQL Type: INT

Regional Settings:

Color of Negative: —

Decimal precision: 0

Complete with Zeros:

- **ata Type** : Define the type of field for the application.
- **ield Mask**: In this field, you can configure a display mask according to the table given in Applications> Grid > Fields> Text.
- **QL Type** : Informs the data type of field in the database.
- **egional Settings** : Allows to apply the Regional Settings to format of the fields. When not enabled, you will view attributes for set yourself. You can change of the Regional Settings in **Locales** > **Regional settings**. For more information [click here](#).
- **olor of Negative**: In this field, you will inform a color in hexadecimal. (Example: # 000000)
- **ecimal Precision** : Number of decimal places. This value is part of the field size.
- **omplete with Zeros** : Defines the decimal places that will or not be completed zeros to the right. Example: The value is 125,50, but the decimal precision is 3. If the option is enabled, then the value will 125,500.

General Setting - Percent Fields

Properties | General Settings | Lookup settings

Data Type: Percent

Related Field:

Field Mask:

Repeat value:

SQL Type: INT

Regional Settings:

Color of Negative: —

Decimal precision: 0

Complete with Zeros:

- **ata Type** : Define the type of field for the application.
- **elated Field**: Field that contains the values for the calculation of the percentage.
- **ield Mask**: In this field, you can configure a display mask according to the table given in Applications> Grid > Fields> Text.
- **QL Type** : Informs the data type of field in the database.
- **olor of Negative**: In this field, you will inform a color in hexadecimal. (Example: # 000000)

General Settings - Date/Time Fields

Properties | **General Settings** | Lookup settings

Data Type: Date and Time

SQL Type: VARCHAR

Regional Settings:

Display: ddmmyyy hhmss

Internal Format:

- **ata Type** : Define the type of field for the application.
- **QL Type** : Informs the data type of field in the database.
- **egional Settings** : Allows to apply the Regional Settings to format of the fields. When not enabled, you will view attributes for set yourself. You can change of the Regional Settings in **Locales** >**Regional settings**. For more information [click here](#).
- **isplay**: Offers a series of formats predefined for displaying dates.
- **nternal Format**: Defines the internal format of the fields (for database storage) date, time, and datetime if the field type in SQL is varchar type.

General Settings - Bar code Fields

Properties | **General Settings** | Display Settings

Data Type: Bar code

Label: barcode

Height: 0

Width: 0

SQL Type: Text

- **ata Type** : Define the type of field for the application.
- **eight**: Sets the height in pixel of the image display. Must contain only numbers.
- **idth**: Sets the width in pixel of the image display. Must contain only numbers.
- **QL Type** : Informs the data type of field in the database.

General Settings - QRCode Fields

Properties | **General Settings** | Display Settings

Data Type: QRCode

SQL Type: Text

Level of error correction: 7%

Image Size: 42x42 px

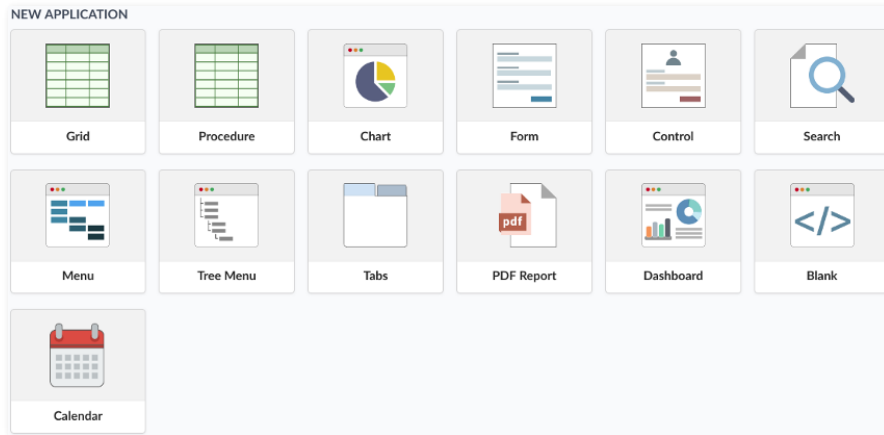
Margin: 1

- **ata Type** : Define the type of field for the application.
- **QL Type** : Informs the data type of field in the database.
- **evel of error correction**: Codewords are 8 bits long and use the Reed-Solomon error correction algorithm with four error correction levels. The higher the error correction level, the less storage capacity.
- **mage Size**: Lets you set the display size of the QR Code by selecting one of the available sizes.
- **argin**: Lets you set the margin width around the QR Code by selecting one of the available sizes.

New Application

The creation process of one application is simple enough and intuitive.

You just need to select which application you want to generate in the applications available in Scriptcase, That it will be generated automatically.



Application Data

When the application is selected, the following screen will be displayed below.

In this table you need to select the connection that will be used in the application. The tables will be loaded and displayed in the field **Table** when the connection is selected.

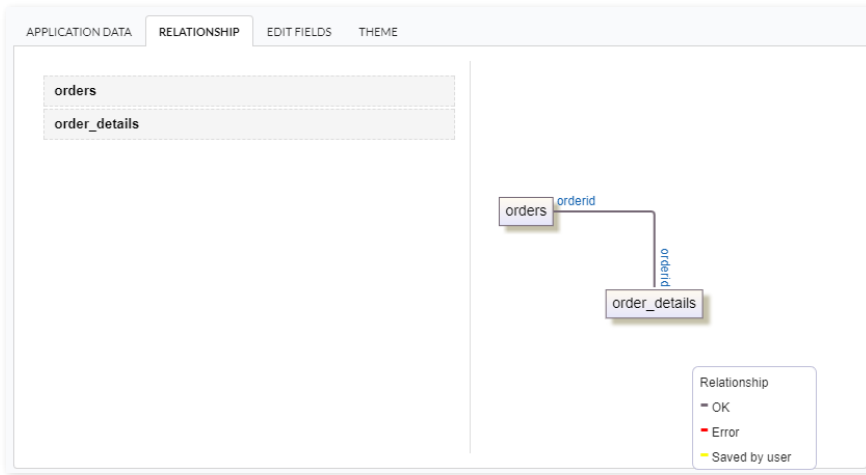
It is possible to select more than one table in the application creation.

After selecting the table and fields that should be part of the application, the name of the application will be formed by the type of application + name of the table. **Ex: grid_orders** However it is possible to change the application's name to any other desired name.

- **connection** - It defines the used connection to create the application. All the project's application will be displayed.
- **ables** - It defines the used tables in the application. (Form and Calendar can only use one table).
- **ields** - It defines the fields that will be part of the applications.
- **QL Select Statement** - Display the select statement created after selecting the tables and fields. This field allows also to insert a previously created SQL, if the used tables are available in the database selected in the connection.

Relationship

When select two or more tables, the tab "Relationship" will be displayed. In this tab we can see the relationship created between the tables, where we can edit the related fields.



When we click in a link, in the screen above, it will displayed the related field's edition form, as you can see in the image below.

Edit Fields

In this tab will be displayed the fields that are going to be created in the application, also its labels and datatypes. We can make define some settings here like changing its labels and datatypes.

Fields	Label	Datatype	Search
id	Id	Integer	<input checked="" type="checkbox"/>
title	Title	Text	<input checked="" type="checkbox"/>
description	Description	Text	<input checked="" type="checkbox"/>

- **ields** - Name of fields stored in the database.
- **abel** - Name of fields displayed in the generated application.
- **atatype** - Field's datatype.
- **earch** - It defines the available fields in the search.

Theme

In this screen you select the theme that the application will use. The default project's theme, that can be checked and changed in **Project > Properties**, will be selected automatically.

- **heme** - List of available themes in the project.

After all the settings are done, click in the button **Create**. In the next screen the settings of the created application will be displayed.

This interface is useful for editing the main display settings of the app.

SETTINGS	
ATTRIBUTE	VALUE
Friendly URL	<input type="text"/>
Line break in title	<input type="checkbox"/>
Horizontal Alignment	Center ▾
Margins	<input type="text"/> Up <input type="text"/> Down <input type="text"/> Right <input type="text"/> Left
Table Width	<input type="text" value="100"/>
Table Width Unit	Percent ▾
Table Columns	Automatic ▾

Form Settings Interface.

Friendly URL

Allows to define a URL for the application different from the application name. You can use the following chars (a-z, A-Z, 0-9, -). You still can change it on the initial screen of the home project, through the "Friendly URL" column in the apps list.

Line break in title

Use it if you want to break the line on the field titles.

Horizontal Alignment

Allows you to set the alignment of the application on the page.

Margins

Allows to define the margins of the application (Right, Left, Up and Down) in pixels.

Table Width

The width of the form table. Scriptcase uses Plain HTML to generate applications by using tables lines and cells.

Table Width Unit

Measurement unit for the table width defined in the previous option, being: percentage, pixel, or automatic.

Table Columns

This parameter defines the column (fields) width of the table (application).

Labels width

When the previous option is set to "Provided" you must inform the width of the labels here.

Layout and Behavior

This interface allows setting the behavior of the app.

LAYOUT AND BEHAVIOR	
ATTRIBUTE	VALUE
Automatic tab	<input type="checkbox"/>
Highlight Text on Focus	<input checked="" type="checkbox"/>
Use Enter to	▾
Field with Initial Focus	▾
Highlight Field with Error	<input type="checkbox"/>
Use a template from the HTML Editor	<input type="checkbox"/>

Layout and Behavior configuration Interface.

Automatic tab

Changes the focus to the next field when the amount of characters reaches its defined limit.

Highlight Text on Focus

Highlights the field when selected.

Use Enter to

Allows to use the "Enter" key to pass the focus to the next field.

Field with Initial Focus

Determines the field starts with focus when accessing the application. This option doesn't work with fields that contain a watermark.

Highlight Field with Error

Focus the field with the error when submitting the form.

Use a template from the HTML Editor

Allows to use the TinyMCE editor. You can edit and create your HTML Templates.

This interface is useful for editing the field settings and their position to display.

Edit Fields

1	Fields	Label	Datatype	New	Update	Read-only	Required	PK	DB value (Insert)	DB value (Update)
5	PAGE: PAG1									
6	BLOCK: FORM_CUSTOMERS									
	customerid	Customerid	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
	companyname	Companyname	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	contactname	Contactname	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	contacttitle	Contacttitle	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	birthdate	Birthdate	Date	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	country	Country	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
7	PAGE: FIELDS NOT SHOWN									
	regionid	Regionid	Integer	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	stateid	Stateid	Select	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	city	City	Text	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	address	Address	Text	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

Edit Fields Configuration.

Fields

It allows accessing the field settings (a pencil icon on the left). You can change the field position by dragging them to the desired position. Drag a field to "fields not displayed" if you don't want it in the app.

Label

It defines the title of the field in the app. For example: if the field name in the database is fld_txt_customer_name, you can display the label "Customer Name".

Data type

It informs the data type of the field.

New

It defines if the field is available when inserting new records.

Update

It defines if the field is available when updating records.

Read-Only

It defines the field as a label. The user can't change its value.

Required

It defines if the field must contain a value.

PK

It defines the Primary Keys fields.

DB value (Insert)

Defines a default value for the field when inserting a new record, like an auto-increment, Date, DateTime, or IP.

DB value (Update)

Defines a default value for the field when updating a record, like an auto-increment, Date, DateTime, or IP.

Page

It shows the pages available in the application. All apps have a page, at least. Each page contains one or more blocks.

Blocks

It shows the blocks available in the application. Blocks contain fields. All apps have a block, at least. A block is displayed if it contains one or more fields.

Page Fields Not Shown

Here we can see the fields that are not in the application.

Observe that you can drag any line to the desired position, blocks, and page. Pages contain Blocks, and Blocks contain fields.

Display Settings

Settings of messages display.

DISPLAY	
ATTRIBUTE	VALUE
Markers positioning	Right
Display message	<input checked="" type="checkbox"/>

Display Interface.

Attributes ### Markers positioning Set the position of the markers that indicates required fields.

Display message
Set it if you want to display the message of the required field.

This interface allows to sort the fields positioning.

Select the fields that will be displayed.

orderid customerid employeeid orderdate requireddate shippeddate shipvia freight priceorder shipcountry shipregion shipstate shipcity shipname shipaddress shippostalcode	➤ ➤ ➤ ➤ ➤	Pag1 form_orders orderid customerid employeeid orderdate requireddate	⤴ ⤴ ⤴
--	-----------------------	---	-------------

Save Restore

Field Positioning

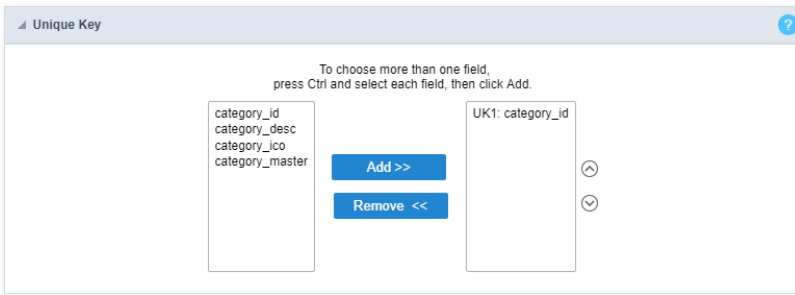
The left box has a list of all fields available in the application. Here you must select which fields you want to display in the app.

The right box contains the selected fields displayed in the application.

Use the arrows buttons between the boxes to select or deselect a field.

With this interface, you can inform which fields shouldn't have repetitions in the database. The app will return a message if the user tries to insert a repeated value into the field.

If you want to combine two or more fields as unique, select the fields by holding the CTRL key of your keyboard and then click on the "Add" button.



The screenshot shows a window titled "Unique Key" with a help icon. Inside, there is a list of fields on the left: "category_id", "category_desc", "category_ico", and "category_master". In the center, there are two buttons: "Add >>" and "Remove <<". On the right, there is a list box containing "UK1: category_id" and two scroll arrows.

Unique Key

To choose more than one field,
press Ctrl and select each field, then click Add.

category_id
category_desc
category_ico
category_master

Add >>

Remove <<

UK1: category_id

Unique key configuration Interface.

The application toolbar has two segments: Top and Bottom, in a way that is possible to define to display buttons into both areas. Those areas work independently, allowing them to display the same button, for example.

It's also possible to select the buttons and their position if the application is running on a mobile device.

Toolbar

Desktop

Here we must inform the toolbar settings for the "Classic Web Version" mode and which buttons are available in the application when accessed from a **Desktop** environment.

Mobile

Here we must inform the toolbar settings for the "Mobile Version" mode. That is which buttons are available in the application when accessed from a **Mobile** dispositive.

Toolbar

Configure the toolbar below for a "classic web version " and also for a "mobile version".

Desktop

Mobile

Top Toolbar

Others
 QuickSearch
 Dynamic Search
 Languages
 Themes
 HelpCase
 Rows Counter
 Jump to
 Copy
Navigation
 Navigation by page
 First
 Previous
 Next

Left
 QuickSearch
 Dynamic Search
 Center
 Insert
 Cancel
 Update
 Delete
 Right
 Exit

Group

Add
Edit
Delete

Bottom Toolbar

Others
 QuickSearch
 Dynamic Search
 Languages
 Themes
 HelpCase
 Rows Counter
 Jump to
 Copy
Navigation
 Navigation by page
 First
 Previous
 Next

Left
 Jump to
 Center
 First
 Previous
 Navigation by page
 Next
 Last
 Right
 Rows Counter

Group

Add
Edit
Delete

Navigation:

Buttons relative to the navigation of the application.

Next	Move to the next page that can be a single record or a list of records.
Previous	Returns to displays the previous page records or a single record.
First	Move to the First page or record
Last	Move to the Last page or record
Exit	Close the application
Navigation by page	Displays a "page-number" navigation bar. Example: 1 2 3 4 5

Export:

The options available to export the Records. Scriptcase generates the following export formats for Forms:

PDF	Generates all the data of the application in a PDF format.
Print	Creates an HTML with the records ready for printing.

Update:

The CRUD options available in the Form.

Insert	Inserts the record into the database.
Update	Saves the changes made in a record.
Delete	Deletes the selected record.
Cancel	Cancel the changes made in a record before the insertion.

Others:

Other options available in the Form application.

Jump To	Move to the informed page or record.
Copy	Copy the current record data to another one.
Quick Search	Perform a quick search in the records of the application.
Dynamic Search	It displays the fields of the search to filter the records.
Languages	Displays a Combobox with the languages available in the project properties.
Themes	Displays a Combobox with the languages available in the project properties.
Rows Counter	Displays the number of records retrieved in the application.
HelpCase	Displays a button to open the help page.

Separator:

-----	Displays a line separating the buttons.
-------	---

Toolbar Mobile

Contains the same options as the **Desktop** version, adding only the item "Copy from Desktop", on which, when clicked, it copies the items from the **Desktop** toolbar to the **Mobile** toolbar.

Buttons Group

The **Group** option allows you to group a set of buttons of the application toolbar to display them as a dropdown, for example.

Add
Add a new group of buttons.

Edit

Edit an existent group of buttons.

Delete

Delete the selected group of buttons.

When you press the **Add** or **Edit** option, you can see the settings to configure the grouper:

Edit

DISPLAY AS

DROPDOWN LIST THEME NAME

LABEL HINT \ TITLE

IMAGE BUTTON TYPE

DISPLAY DISPLAY POSITION

Display As

Allows displaying the group button as **Dropdown** or **Side by Side**.

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

Customer ID	Company Name	Phone	Credit Limit
dos		30074321	\$3,367.41
arilla		55554729	\$7,371.95
		55553932	\$6,757.53

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

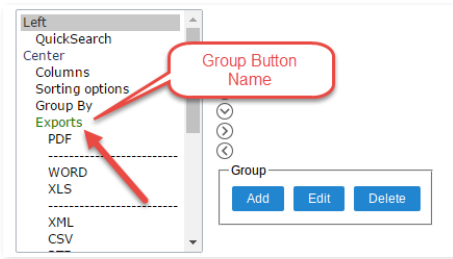
Exit

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print

Exit

After creating a button group, you need to move the grouped buttons below of the Button Group and then move them to the right. Like the image below:



Buttons Settings

Button	Label	Hint
QuickSearch		
Dynamic Search		
Insert		
Cancel		
Update		
Delete		
Exit		
Jump to		
First		
Previous		
Navigation by page		
Next		
Last		
Rows Counter		
New		

Button:

It displays the buttons available in the application.

Label:

Allows defining the labels of the buttons to display for the users.

Hint:

Allows defining the buttons hint that to display for the users.

Application Hotkeys

Scriptcase allows creating shortcut keys to your applications. You can select a predefined template or create specific actions for an application.

VALUE	DESCRIPTION				
<input checked="" type="checkbox"/> toggleHotkeys();	Define if the application will use hotkeys				
SC_DefaultHotkeys	Select the hotkey template from previously created schemas				
<input type="button" value="Clear"/> <input type="button" value="+"/>					
<table border="1"> <thead> <tr> <th>ACTION</th> <th>KEYBINDING</th> </tr> </thead> <tbody> <tr> <td colspan="2" style="text-align: center;">No hotkeys configured</td> </tr> </tbody> </table>		ACTION	KEYBINDING	No hotkeys configured	
ACTION	KEYBINDING				
No hotkeys configured					
<input type="button" value="Clear"/> <input type="button" value="+"/>					

Use hotkeys

Defines if the application uses hotkeys. When you enable this option, the default shortcut keys settings are disabled.

Hotkeys templat

Select the [hotkey template](#) previously created.

Action

Selects the triggered action when pressing the selected key.

Keybinding

Selects the keys responsible for executing the chosen action.

Add "+"

Adds a new action on the keys list.

Clear

It clears the selected hotkeys preference.

Options

ATTRIBUTE	VALUE
Format Row Counter	<input type="text"/>
The number of links displayed	<input type="text" value="5"/>
Help by Block	<input type="checkbox"/>
General Help	<input type="checkbox"/>
Toolbar buttons	<input type="text" value="A DIV below the toolbar. ▾"/>

Format Row Counter:

Allows defining the format of the row counter displayed on the application.

Example: (1 to 10 of 200)

The Number of Links Displayed:

Defines the number of links per page, when the navigation option is disabled.

Help by Block:

Indicates if the helps messages from relatives fields are grouped by block. In each field, we can define a help text. With this option activated, it shows up an icon in the block title bar to call the help page.

General Help:

The General Help "consolidates" all the fields help pages in a single page, putting an icon in the toolbar to call the help page.

Toolbar Buttons:

The type of view for the Toolbar buttons (A DIV below the toolbar, Modal).

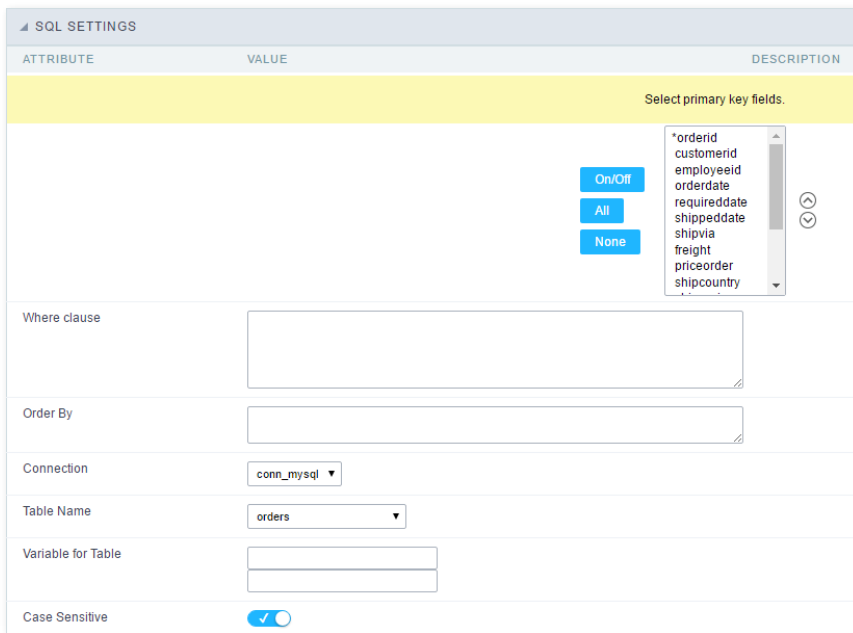
title: Print settings

PRINT	
ATTRIBUTE	VALUE
Show Print button	<input checked="" type="checkbox"/>
Format of date display in printing	<input type="text"/> ?
Displays title in the printing	<input type="text" value="(lang_events_order)"/>
Additional fields	<div style="display: flex; align-items: flex-start;"><div style="border: 1px solid #ccc; padding: 2px; margin-right: 5px;"><div style="text-align: right; padding-right: 5px;">description ▲</div><div style="text-align: right; padding-right: 5px;">end_date</div><div style="text-align: right; padding-right: 5px;">recurrent</div><div style="text-align: right; padding-right: 5px;">period ▼</div></div><div style="margin-right: 5px;"><input type="button" value="Add >>"/></div><div style="border: 1px solid #ccc; width: 40px; height: 20px; margin-right: 5px;"></div><div><input type="button" value="Remove <<"/></div></div>

Print export settings.

- **isplay print button**: This option enable a button within the application to print the calendar.
- **rint format**: It changes the display data format for printing.
- **isplay title for printing**: Displays the calendar title for printing.
- **dditional fields**: Additional fields to be displayed for printing.

This interface allows configuring the related database settings, such as the Primary Key, Filters, Sorts.



ATTRIBUTE	VALUE	DESCRIPTION
Select primary key fields.		
<div style="display: flex; justify-content: center; gap: 10px;"> On/Off All None </div> <div style="border: 1px solid gray; padding: 5px; margin-top: 5px;"> *orderid *customerid *employeeid *orderdate *requireddate *shippeddate *shipvia *freight *priceorder *shipcountry </div>		
Where clause	<input type="text"/>	
Order By	<input type="text"/>	
Connection	conn_mysql ▼	
Table Name	orders ▼	
Variable for Table	<input type="text"/> <input type="text"/>	
Case Sensitive	<input checked="" type="checkbox"/>	

SQL configuration Interface.

Select primary key fields

It lets you define the Primary key of the Form. ScriptCase already identifies Primary Keys, but you can manually inform or change it by using the buttons beside the fields list. See how the buttons work:

- **n/Off** : Adds or Removes the attribute that defines the primary key for the field. The primary key fields have an asterisk beside their names.
- **ll** : Defines all fields as Primary Keys.
- **one** : Defines none fields as Primary Keys.
- **orting Button** : These are the arrows on the right side of the Combo box. It allows ordering the fields of the Primary Key, placing it in the desired order. To order them, click on the field and use the arrows to move it.

Where clause

It allows adding a WHERE clause to filter the SQL records.

Order By

It allows adding an ORDER BY clause to determine the order to display the records. By default, it uses the primary key to sort the records.

Connection

It allows defining the database connection of the application. You can change the connection to another one that has the same table.

Table Name

It informs the database table used in the Form.

Variable for Table

It allows to use a variable to change a part of the string containing the table name.



Variable for Table Configuration.

Fill the first input with the name of the variable (replace to). The second input you should fill with the part of the name of the table you want to replace (replace from).

Case sensitive

It defines if the database connection uses case sensitive or not.

This interface allows configuring a Form application to execute the Stored Procedures from your database for Insert, Update and Delete records in the form. You don't need to use the three options simultaneously. If a Procedure option is not enabled, it continues to work as default by using the INSERT, UPDATE and DELETE commands.

See below, the step-by-step to enable a Stored Procedure for inserting records into the database.

Form Procedures ?

Enable INSERT Procedure.

Enable UPDATE Procedure.

Enable DELETE Procedure.

Next

Form Stored Procedures Interface.

- et's check the first option: "Enable INSERT Procedure". Then, inform the Name of the Procedure and the number of parameters.

Form Procedures ?

Enable INSERT Procedure.

Procedure Name

Parameters Qty

Enable UPDATE Procedure.

Enable DELETE Procedure.

Next

Defining parameters for the Procedure.

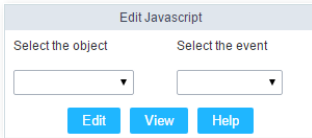
- ow associate the fields for each parameter of the Stored Procedure and the type of parameter (Input or Output). Click on the "Save" button to finish.

Parameters Number	Parameters	Parameters Type	Procedure variable name
Param1	<input type="text"/>	<input checked="" type="radio"/> Input <input type="radio"/> Output	
Param2	<input type="text"/>	<input checked="" type="radio"/> Input <input type="radio"/> Output	
Param3	<input type="text"/>	<input checked="" type="radio"/> Input <input type="radio"/> Output	
Param4	<input type="text"/>	<input checked="" type="radio"/> Input <input type="radio"/> Output	

Save Cancel

Passing the values to the Stored Procedure Parameters.

To coding with JavaScript in Scriptcase, we must associate a JavaScript event to a form field.



Edit JavaScript Interface.

Select the object

This Combo box displays the fields of the form application, and also the form itself as an object

Select the event

Use it to associate the event with the selected field to apply the JS code. View the events available:

- **nClick**: Acts when clicking on the field.
- **nBlur**: Acts when the focus leaves the object.
- **nChange**: Acts when the focus leaves the object, and there are changes in the value.
- **nFocus**: Runs when the focus gets in the object.
- **nMouseOver**: Runs when the mouse pointer hovers the object.
- **nMouseOut**: Runs when the mouse pointer moves out the object.

Events related to the Form

The events below are associated directly with the form object.

- **nLoad**: This event runs when the page is loaded, also when clicking on the navigating buttons.
- **nSubmit**: This event runs when clicking on the “New”, “Save”, and “Delete” buttons.

Edit JavaScript

*Select the object and event, then click on the Edit button. It opens a page to inform custom JavaScript routines and

```
![Edit JavaScript Interface][javascript_edicao_code]
*Edit JavaScript Interface*
```

> The JavaScript language doesn't have the same behavior in all the browsers available. A tip is to test running the

- **nClick Example**
 - When clicking on a field of type radio, you can enable or disable form fields according to the selected value.

```
if(document.F1.gender[0].checked){
  document.F1.maternity.disabled = false;
  document.F1.maternity.style.background='FFFFFF'
}
if(document.F1.gender[1].checked){
  document.F1.maternity.disabled = true;
  document.F1.maternity.style.background='FCEEB3'
}
```

To access the values of a radio field, you need to use the index.

- **nBlur Example**
 - You can define a warning for the field “weekly_work_time” when the focus is getting out it.

```
if (document.F1.tp_point[0].checked && document.F1.weekly_work_time.value > '20')
{
  alert("The work time exceeds the limit allowed");
  document.F1.weekly_work_time.value = "";
  document.F1.weekly_work_time.focus();
}
```

- **nChange Example**
 - When modifying the “Salary” of an employee and leaving the field, we’ll check if his “position” is “gardener”.

```
if (document.F1.salary.value > 2000.00 && document.F1.position.value == 'gardener'){
  alert('When I grow up, I want to be a gardener');
}
```

- **nFocus Example**
 - After informing the purchase value and selecting the payment method in a Select object “Select: pay_method”, the JavaScript code below calculates the value of the purchase.

```
if (document.F1.pay_method[document.F1.pay_method.selectedIndex].text == 'Money')
{
  document.F1.total.value = document.F1.purchase_value.value;
}
```

- **nMouseOver Example**

- You can change the style (background color, font and font color) when the mouse hovers the field.

```
document.F1.field_name.style.backgroundColor = "0FFCCA"
```

- **nMouseOut Example**

- Resets the background color when the mouse moves out from the field.

```
document.F1.field_name.style.backgroundColor = "FFFFFF"
```

This feature allows linking the tables that contain relationships, dependencies, like the tables orders and order_details. This way, it's possible to delete a record in the table orders, and it automatically deletes all the details for that order as well. Below let's see a practical example of this feature.

1. firstly, you must define a new dependency. Then you must select the dependent table. Click on the button New Dependency to start.

Creating a new Dependency.

1. you must inform the fields amount is related between both tables. In this example, we have only one field.

Dependency Table configuration.

1. then select the related key fields: order_details->Orderid - orders->Orderid.

Dependency Key Fields Configurations.

1. now we need to select behaviour for the application. Then click on the button Generate Scripts to define the Dependency Rule.

Configuring dependency Rules.

Security		
ATTRIBUTE	VALUE	DESCRIPTION
Use Security	<input type="checkbox"/>	Application needs validation to run. Read about the sc_apl_status macro.
Url output of the security	<input type="text"/>	Uri to be redirected if the user does not have access to this application.
Use Password	<input type="checkbox"/>	Enter password required to run the application.
Request password just once	<input type="checkbox"/>	Request password just once per session without asking for password every time you open the application in the same session.
Enable direct call by URL	<input checked="" type="checkbox"/>	Allow an application to be called by typing the URL directly in the browsers.

Application Security Configuration

Use Security

Turns the application accessible just when using the macro sc_apl_status or using the security module. When it is enabled, the application displays a message "User not Authorized" if the user tries to access it directly.

Url output of the security

It redirects the application to another URL after displays the warning "User not Authorized".

Use Password

Turns the application accessible just by using a password.

Request password just once

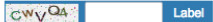
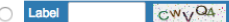
Request the password once per session, not needing to inform the password every time you access the application.

Enable direct call by URL

Allows to call the application directly from the URL in the browsers.

Allows calling the application directly from the URL in the browsers. In the development environment, you can disable those options for testing, it avoids requiring the login every time you run the application. To disable the security and password in development, you must access Options > My ScriptCase and uncheck the options: Enable Use of Security, Enable Use of Password.

Captcha

Captcha		
ATTRIBUTE	VALUE	DESCRIPTION
Use Captcha	<input type="checkbox"/> Captcha	Enables Captcha function for this application.
Number of Characters	<input type="text" value="4"/>	Number of characters
Character List	<input type="text" value="ABCDEFGHIJKLMNOPQRSTUVWXYZ01234"/>	Character list.
Label	<input type="text" value="{lang_othr_cptc_lbel}"/>	Message shown near Captcha.
Error message	<input type="text" value="{lang_othr_cptc_errm}"/>	Captcha error message.
Height	<input type="text" value="65"/>	Picture height in pixels.
Width	<input type="text" value="175"/>	Picture width in pixels.
Font Size	<input type="text" value="20"/>	Font size of the captcha.
Reload	<input type="checkbox"/>	Add a button to reload Captcha.
Select one layout		
<input checked="" type="radio"/>  <input type="radio"/> 		

Use Captcha

Defines if the application uses Captcha.

Number of Characters

Amount of characters in the Captcha image.

Character List

List of characters used in the Captcha.

Label

The message displayed for the Captcha.

Error message

Captcha error message.

Height

Height of the image generated by the Captcha (in pixels).

Width

Width of the image generated by the Captcha (in pixels).

Font Size

Size of the font used in the image generated by the Captcha (in pixels).

Reload

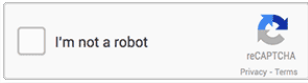
Display the refresh button in the Captcha.

Select one layout

It offers various layouts for the Captcha display.

reCAPTCHA

reCAPTCHA is an API provided by Google for forms. It adds security, preventing automatic submission of forms through robots.

reCAPTCHA sample:

1. first, we must request an API Key to activate reCAPTCHA into a Scriptcase application by following the steps below:

To get a **Site key** and **Secret Key** go to the link: <https://www.google.com/recaptcha/admin#list>. See the image:

Register a new site

Label

For example, example.com: Comments page

Choose the type of reCAPTCHA ?

reCAPTCHA V2
Validate users with the "I'm not a robot" checkbox.

Invisible reCAPTCHA
Validate users in the background.

reCAPTCHA Android
Validate users in your android app.

Domains
(one per line)

For example:
example.com
example.net
example.org

Accept the reCAPTCHA Terms of Service.
By accessing or using the reCAPTCHA APIs, you agree to the Google APIs Terms of Use, and to the Additional Terms below. Please read and understand all applicable terms and policies before accessing the APIs.
» reCAPTCHA Terms of Service

Send alerts to owners ?

Register

Label

It is a project identifier to create the reCAPTCHA keys.

Choose the type of reCaptcha

We must choose the option **reCAPTCHA V2**.

Domains

We can insert multiple domains (one per line) to limit the API uses.

1. hen, we need to accept the Terms of Service ("Accept the reCAPTCHA Terms of Service").
2. hen clicking on **Register**, the page refreshes and shows the integration of reCAPTCHA information. There we can get the **Site Key** and **Secret Key**:

ⓘ Adding reCAPTCHA to your site

▼ Keys

Site key Use this in the HTML code your site serves to users.	Secret key Use this for communication between your site and Google. Be sure to keep it a secret.
<input type="text"/>	<input type="text"/>

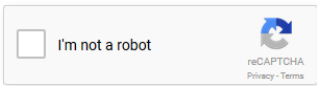
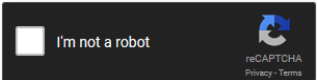
1. ow, we can set the Scriptcase application security:

ATTRIBUTE	VALUE
Use Captcha	Recaptcha
Site Key	
Secret Key	
Theme	light
Type	Audio
Size	Compact
Position	Center

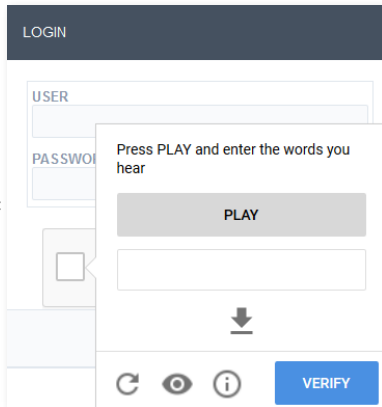
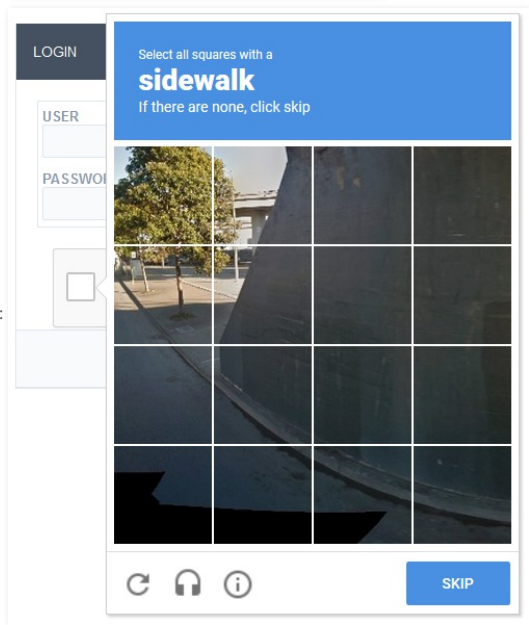
Site Key
Key generated by Google after reCAPTCHA project creation.

Secret Key
Key generated by Google after reCAPTCHA project creation.

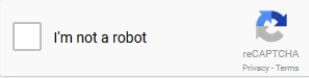
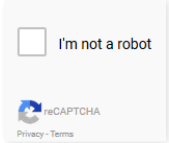
Theme
Defines the reCAPTCHA color. There are two options:

- **light** : 
- **dark** : 

Type
The type of reCAPTCHA. There are two options:

- **audio**: 
- **image**: 

Size
The size of reCAPTCHA. There are two options:

- **normal**:  A rectangular widget with a light gray background. On the left, there is a small square checkbox followed by the text "I'm not a robot". On the right, there is a circular reCAPTCHA logo with the text "reCAPTCHA" and "Privacy - Terms" below it.
- **compact**:  A rectangular widget with a light gray background. On the left, there is a small square checkbox followed by the text "I'm not a robot". On the right, there is a circular reCAPTCHA logo with the text "reCAPTCHA" and "Privacy - Terms" below it.

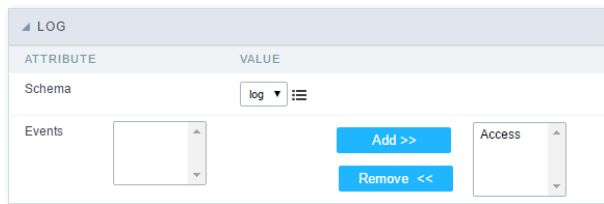
Position

Here we can define the reCAPTCHA component alignment:

- **eft**: Position the reCAPTCHA component at the left.
- **enter**: Position the reCAPTCHA component at the center.
- **ight**: Position the reCAPTCHA component at the right.

This interface allows you to define a Log schema to the app. The Log schema tracks what the end-users are doing in the app. Those are the events it can tracks: Access, Insert, Update, and delete.

To learn how to create a Log schema [see here](#).



The screenshot shows a configuration window titled "LOG". It features a table with two columns: "ATTRIBUTE" and "VALUE". The "Schema" attribute has a dropdown menu currently set to "log". Below the table, there are two empty list boxes: "Events" on the left and "Access" on the right. Between these two boxes are two blue buttons: "Add >>" and "Remove <<".

Application Log configuration

Schema Select an existing log schema for the application.

Events You can select just specific events to save in the log.

Calendar fields

This interface displays the basic Calendar field settings.

CALENDAR FIELDS

Id *	<input type="text" value="id"/>	Recurrence	<input type="text"/>	Enabled	<input type="text" value="Y"/>
Title *	<input type="text" value="title"/>			Disabled	<input type="text" value="N"/>
Start date *	<input type="text" value="start_date"/>	Period	<input type="text"/>	Daily	<input type="text" value="D"/>
Start time	<input type="text" value="start_time"/>			Weekly	<input type="text" value="W"/>
End date	<input type="text" value="end_time"/>			Monthly	<input type="text" value="M"/>
End time	<input type="text"/>			Annual	<input type="text" value="A"/>
		Format Time	<input type="text" value="13:00"/>		
		Start Mode	<input type="text" value="Month"/>		

Calendar fields interface.

1. [Home](#)
2. Applications

General Overview

The applications can have two types of fields.

Table Fields

Columns of the table connected to the application and their data types.

Example:

In a table with the following structure:

```
CREATE TABLE `orders` (
  `orderid` int(11) NOT NULL,
  `customerid` varchar(5) NULL DEFAULT NULL,
  `employeeid` int(11) NULL DEFAULT NULL,
  `orderdate` date NULL DEFAULT NULL,
  `requireddate` date NULL DEFAULT NULL,
  PRIMARY KEY (`orderid`) USING BTREE
);
```

These are the listed fields on left menu **Fields**:

Virtual Fields

Fields created inside application to assist in the development process.

How to create a virtual field

1. Inside an application, access the left menu **Fields** and click on **New Field** option.
2. Choose how many fields do you want to create:

1. Create the fields and define type, name and label for each one:

- **Type:** Data type of the field.
- **Name:** Internal name of the field. Used to identify the field on events and Scriptcase interface.
- **Label:** Title of the field displayed on the application.

Scriptcase **doesn't insert** virtual fields on the table after its creation.

Types of fields by applications

Application Table Fields Virtual Fields

Grid	<input type="text"/>	<input type="text"/>
Procedure	<input type="text"/>	<input type="text"/>
Chart	<input type="text"/>	<input type="text"/>
Form	<input type="text"/>	<input type="text"/>
Control	<input type="text"/>	<input type="text"/>
Search	<input type="text"/>	<input type="text"/>
PDF Report	<input type="text"/>	<input type="text"/>
Calendar	<input type="text"/>	<input type="text"/>

The applications **Menu**, **Tree Menu**, **Dashboard** and **Blank** were not listed because they haven't manage fields.

- 1. [Home](#)
- 2. [Applications](#)
- 3. [Calendar](#)
- 4. [Text](#)

Text

General Settings

This type of field allows the developer to create quickly inputs to insert and update data, where the final user can inform its data to be allocated in its database.

Type of Data

Defines the type of the field in the application. In this case we should select **Text**.

Label

Defines the title that will be displayed in the field when the application is executed. The terminology used in the interface is fundamental to the system have a good usability, who should use names and familiar terms to the final user of the application, instead of using terms from the system.

For example, this text field that has the name **customerid**, the client would have a much better understanding of the functionality of the field when we define the label as **Customer Name**.

Not only a fixed text, the **Label** attribute allows the use of langs to define the field's title, making it possible to use your application in a multi language project.

Watermark

Informing a text to the **Watermark** it will be displayed in the input a text as an example that can be informed in the field. The result after set will be this:

Initial Value(type)

Allows the initial definition to the field when the form is in insert mode. It is possible to choose between two options:

Defined Value: When this option is selected, the Initial Value attribute will be available, where we should inform the field's initial value. For example, my initial value is **Arlindo**, when a new register is inserted, the field **Seller Name** will be initialized as Arlindo.

System Data: When this option is selected, the initial value will be the actual date of your computer's system.

Amount of Characters

Allows to set the width of the text field's input that varies with the amount of characters informed. Although, if the amount of characters typed are greater than the setting, the text will be pushed to the left, to keep the maximum amount of characters as defined.

Show HTML Content

When this option is active every HTML, CSS and JavaScript content that are in the database will be displayed with the main value.

Validation Image

When this option is active, a image will be displayed next to the informed field if the field is according to the settings defined by the developer.

In the example below, the field was set to receive at least 5 characters, see what happens when informed only 4 characters:

However if informed 5 or more characters the field will be displayed as:

Password Field

When this option is active, the text field will be converted to the format used in password fields. For example:

Save Variable

Allows to save a session variable(global variable) with the field's value, to be used in others applications.

For example, in the login form the username can be saved in session and displayed on the header of others applications.

Variable Name

In this attribute we should define the name of the session variable, active in the previous item, that will receive the field's value.

We should inform only the variable's name, - **var_rating**.

The method to use its value is [global variable](#).

Field Mask

Defines the field mask. There are two types of mask described in the table below:

Character Description

- | | |
|---|--|
| X | It is overridden by any characters returned by the database. Filled is required and the value will be completed with leading zeros when there are fewer characters than the size of the mask. |
| Z | It is overridden by any characters returned by the database. Its filling is optional and when there are fewer characters than the mask size, nothing will be done about the characters that are missing. In addition the leading zeros will be suppressed. When used in conjunction with the X necessarily the use of this character is to the left of the mask. |
| 9 | It represents any numeric character (from 0-9) |
| A | It represents an alpha numeric character (A-Z,a-z) |
| * | It represents any alpha-numeric character (A-Z,a-z,0-9)entered by the user. |

For example, it is possible to set the mast to display a telephone number:

It will be show with this format on runtime:

It is also possible to set the field mask like those examples:

Field mask examples:

Telephone number

Field	Mask Input	Typed Value	Formatted Value
Phone Number	+99 99 9999 - 9999	123456789012	+12 34 5678 - 9012
Phone Number	(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
Phone Number	(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
Phone Number	(zz) xxxx - xxxx	12345678	() 1234 - 5678
Software Key (Letters only)	AAAA-AAAA-AAAA-AAAA	QWERTYUIASDFGHJK	QWER-TYUI-ASDF-GHJK
Software Key (Letters and numbers)	xxxx-xxxx-xxxx-xxxx	Q1W2E3R4T5Y6U7I8	Q1W2-E3R4-T5Y6-U7I8
License Plate	AAA - 9999	QWE1234	QWE - 1234
Scriptcase Serial Key	A999A999A999-**	D111H111G111DG2P	D111H111G111-DG2P
Multiple masks (Phone Number)	9999-9999;(99)9999-9999; 9999 999 9999		+99 99 9999-9999

Validate with mask

When this option is active it is possible to insert the data with the mask set in the **Field Mask** option.

Complete to the left

When this option is used, the value defined will be added on the left of the information inserted in the field if the value is lower than the maximum set in the **Maximum Size** option.

Field size in database

Defines the field's size related to the size set in the database. This value is already set automatically by default when the application is generated.

Hidden Field

When this option is active, the field will be hidden in the application on runtime.

Label Field

When this option is active, the field will be altered to only a label where the info will be displayed, where updates or inserts will not be possible.

Save HTML tags

When this option is active, it allows to HTML tags in the field to be inserted with the data, instead of being interpreted.

Text input in JavaScript

When this option is active, it will be show every JavaScript content inserted in the database with its main value.

This option can only be used using the **Editable grid view**.

Disable Field

Allows the developer to disable the field, so the user can not type a value according with the option defined by the developer.

The available options are: **No** - Initial value of the attribute, this option does not disable the field.

Update Mode - This option only disables the field when editing existing registers.

Insert Mode - This option only disables the field when inserting new registers.

Insert / Update Mode - This option disables the field when editing exciting registers or when inserting new registers.

HTML Type

HTML object used to display the field in the form.

SQL Type

Informs the type of the field in the database.

Values Format

Format of Values Interface

- **Case Settings** : Allows to convert the letters of the field when losing the focus. The options are:
 - **Upper Case** : All in Upper Case.
 - **Lower Case** : All in Lower Case.
 - **Capitalize first word** : Capitalizes the first letter of the word.
 - **Capitalize all words** : Capitalizes the first letter of all the words.
- **Minimum Size** : Allows to define the minimum amount of characters accepted for the field.
- **Maximum Size** : Allows to define the maximum amount of characters accepted for the field.
- **Allowed Characters** : Allows to select a set of characters that can be typed into the field. The options are:
 - **All** : Allows any type of character to be typed into the field
 - **Selected** : Defines a set of characters allowed. Using the configuration below, will be allowed to type letters, numbers and other characters (defined in the attribute More). The letters a, b and c (defined in the Less Attribute) are not allowed.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Calendar
4. Multiple Lines Text

Multiple Lines Text

General Settings

Multiple Lines Text field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Multiple Lines Text , you can inform a Text value to the field in multiple lines.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : you will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Lines** : Allows to define the amount of lines that the field will have at start.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Field size in database** : Determines the field size in bytes. it is used to determine the max size of characters that is allowed to type in.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Save HTML tags** : This option allows to save the HTML content of the field to the database.
- **Text input in JavaScript** : Allows to save JavaScript code informed in the field to the database.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface

- **Case Settings** : Allows to convert the letters of the field when losing the focus. The options are:
 - **Upper Case** : All in Upper Case.
 - **Lower Case** : All in Lower Case.
 - **Capitalize first word** : Capitalizes the first letter of the word.
 - **Capitalize all words** : Capitalizes the first letter of all the words.
- **Minimum Size** : Allows to define the minimum amount of characters accepted for the field.
- **Maximum Size** : Allows to define the maximum amount of characters accepted for the field.
- **Allowed Characters** : Allows to select a set of characters that can be typed into the field. The options are:
 - **All** : Allows any type of character to be typed into the field
 - **Selected** : Defines a set of characters allowed. Using the configuration below, will be allowed to type letters, numbers and other characters (defined in the attribute More). The letters a, b and c (defined in the Less Attribute) are not allowed.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.

- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Calendar
- 4. Integer

Integer

General Settings

Data type:

It sets the application field type. When the field is set as Number, it is permissible to define formatting rules of integers.

Label:

It sets the title that will be displayed in the field when you run the application. The terminology used in the interface is fundamental for your system has a good usability. In this case, we recommend you to use names and terms familiar to the end user of the application, instead of using terms originated in the system.

Watermark

Informing a text for the **watermark** it will be displayed on the input text with an example of what can be reported in the field. The result after that will be configured:

Initial value (type)

Allows setting an initial value for the field when the application is in inclusion. You can choose between two options:

Pre-set value: When you select this option, the Initial Value attribute will be made available in the place where we inform the initial value of the field. For example, my initial value is **Arlindo** when inserting a new record the **seller_name** field will start with Arlindo.

System date: When you select this option, the initial value will be the current system date of your computer.

If you select the type system date, it will be not necessary to fill in the initial value attribute.

Amount of characters

It allows you to set the width of the input text field according to the amount of characters. However, if the quantity entered is greater than the set for the characters, the text will be pushed to the left, in order to ensure the maximum amount of characters set in the option of **Values formatting**.

Validation Image

When you enable this option, an image will be shown next to the field informing whether the field is in accordance with the settings of *minimum size* and *Maximum size* (as you can see in the images below) defined in the Formatting value option.

In the example below, the field was set to receive at least 5 characters, see what happens with the image when it receives only 4 characters:

However, if the value inserted has 5 or more characters the the image will change according to example bellow:

Use slider:

It displays a slider component in the field. So you can increase or decrease the value sliding the cursor. You can also customize the increment value, if it increments the value 1 by 1, 2, 5, 10... N.

Field Mask Input:

It sets the mask input for the field. There are two types of mask as described in the table below:

Character Description

X	It is overridden by any characters returned by the database. Filled is required and the value will be completed with leading zeros when there are fewer characters than the size of the mask.
Z	It is overridden by any characters returned by the database. Its filling is optional and when there are fewer characters than the mask size, nothing will be done about the characters that are missing. In addition the leading zeros will be suppressed. When used in conjunction with the X necessarily the use of this character is to the left of the mask.
9	It represents any numeric character (from 0-9)
A	It represents an alpha numeric character (A-Z,a-z)
*	It represents any alpha-numeric character (A-Z,a-z,0-9)entered by the user.

Mask Input Examples:

Field	Mask Input	Typed Value	Formatted Value
Phone Number	+99 99 9999 - 9999	123456789012	+12 34 5678 - 9012
Phone Number	(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
Phone Number	(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
Phone Number	(zz) xxxx - xxxx	12345678	() 1234 - 5678
Software Key (Letters only)	AAAA-AAAA-AAAA-AAAA	QWERTYUIASDFGHJK	QWER-TYUI-ASDF-GHJK
Software Key (Letters and numbers)	xxxx-xxxx-xxxx-xxxx	Q1W2E3R4T5Y6U7I8	Q1W2-E3R4-T5Y6-U7I8
License Plate	AAA - 9999	QWE1234	QWE - 1234
Scriptcase Serial Key	A999A999A999-**	D111H111G111DG2P	D111H111G111-DG2P
Multiple masks (Phone Number)	9999-9999;(99)9999-9999; 9999 999 9999		+99 99 9999-9999

Validate with Mask

By enabling this option you will be able to enter data according to the mask that was configured in the option **Field Mask** and Scriptcase will validate it.

Record Variable

Allows you to record a session variable with the value of the field ([global variable] [var_glob]), to be used in other applications.

Example, in the login form the username can be saved in session and displayed in the header of other applications.

Variable Name

In this option you must set the session variable name, enabled in the previous item, which will receive the value of the field.

You need to inform only the variable name, for example: **var_seller**.

The recovery of the value is made as [Global Variable][var_glob]{:target='blank'}.

Field size in Database

This option sets the size of the field relative to the size that is configured in the database. By default this value is already configured automatically when the application is generated.

Hidden Field

This option when enabled will hide the field inside the application at the time of execution.

Label Field

By enabling this option, the field will be changed to only one label where the information will be displayed, so it is not possible to make changes or inserts in the field configured as label.

Disable field

Allows you to disable the field, making it impossible for the user to enter a value according to the developer-defined option.

The available options are:

No - initial value of the attribute, this option does not disable the field. **Update mode** - This option disables the field only when editing the records. **Insert mode** - This option disables the field only inserting new records. **Update/Insert mode** - This option disables the field in both editing and inserting new records.

HTML Type

HTML object used to display the field in the application.

Tipo SQL:

Reports the SQL type of the field as it is configured in the database.

Values Format

Use regional settings:

It allows you to apply regional settings to the number fields formatting. When this option is not selected it will be displayed the attributes group, minus sign and negative number format.

Digit grouping:

It allows you to define if the field displays the digit grouping separator.

Maximum Size:

It allows you to define the maximum size that will be used in the field of the application.

Minimum Value:

It allows you to define the minimum value that will be used in the field of the application.

Maximum Value:

It allows you to define the maximum value that will be used in the field of the application.

Grouping:

It allows you to define which character is used to separate thousands. This option is only available when the option to use regional settings is disabled.

Negative sign:

It allows you to define which character will be used for the display of negative numbers. This option is only available when the option to use regional settings is disabled.

Negative number format:

It allows you to define the placement of the negative sign in relation to value. This option is only available when the option to use regional settings is disabled.

Accept:

It allows to define if the field accept only negative values, positive-only or both.

Display Calculator:

This option when enabled will display a calculator to assist the end user to perform calculations.

Use Spin:

This option applies a Spin component to the number field. So the end user can increase or decrease the numbers using it. You will also be able to set a minor or major range for the spin.

Smaller increment for the Spin::

Sets the minimum range of the Spin component.

Greater increment for the Spin:

Sets the maximum range of the Spin.

Database Value

It allows you to define how the value to be written to the database.

Preset value:

It allows you to specify a fixed value or associate it to global variables created inside the project and save it automatically to the database field.

Auto increment (automatic):

It allows you to use the increase generated automatically by the database to fill the field. It is used only when the database field is of type AUTO INCREMENT or similar. For the databases that use string, such as Oracle, PostGres and Firebird, you must inform the name of the string.

Auto increment (manual):

The application itself will simulate an auto increment in the field. So the value of the field will be calculated automatically by the application.

Date of inclusion:

The field will be populated automatically with server date during the inclusion.

Date of update:

The field will be populated automatically with server date during the update.

User IP:

The field will get the IP of the machine that is accessing the application.

Calculated by the database:

The field value is assigned by the database itself. The field will not be used in the process of updating. For example: it will be updated via a Trigger.

Null:

The field will be automatically filled with null.

Lookup Settings

The Lookup is a way used to provide the end user a list of values description/name where the items can be selected and used in the application. For example, for a field sex that in the database is represented with M or F values can be presented as male or female dynamically with the lookup. So you will be able to retrieve these values from a database table or manually, if the reference table does not exist.

Example:

Lookup Methods

There are two lookup methods available:

Manual: To use this method it is necessary to inform the selection conditions manually to the Lookup.

Automatic: To use this method it is necessary to use a lookup query from the database to access the reference table.

Editing-automatic lookup:

Select Command:

It sets the SQL command that will retrieve the value to display inside the Grid field. The command must have the following format

```
SELECT field_to_be_displayed FROM table WHERE key_field = {grid_field}
```

The Grid field must be always referenced between curly braces {}, So at run time, the field braces will be replaced by its value in the table.

Multiple Options:

When the Select command informed return multiple values, the option must be selected.

Delimiter:

It sets the tab of values returned by the Select command, this option should be filled when the "Yes" option is checked in the multiple attribute options.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

Use a specific connection:

This option allows you to select another existing connection in the project, other than the current connection, to select the values for the lookup.

Edit Lookup - Manual:

Lookup used to modify the display of the field with predefined values.

Lookup Types

It sets the operating mode of the lookup can be: single value, multiple values (delimiter), multiple values (position) and multiple values (binary).

Single value: it will be displayed only one value for the field. There is a direct replacement of the value stored in the Bank for a label defined in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value of "M" will be replaced by "Male".

Multiple Values (Delimiter):

By using this option it will be displayed multiple values for the selected field. The values must be separated by a delimiter to be informed. A replacement of parts from a string stored in the field, separated by a delimiter for values contained in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value S;M will be replaced by Sports and Movies.

Delimiter: Character used to separate the values within the same string.

Multiple Values (Position)

Allows you to recover information stored in a single string of the selected field. In order for this information to be retrieved must be informed, in addition to the label, the starting position and the number of bytes that each information occupies within the string.

As an example we will use a string to store Sex, Marital status and Hobby respectively. Sex occupies one byte, Status one byte and Hobby two bytes .

For this purpose we define the list as:

Label	Value	Start	Size
Male	M	1	1
Female	F	1	1
Married	M	2	1
Single	S	2	1
Study	ST	3	2
Sportes	SP	3	2
Reading	LE	3	2

Example: the string MMST would be presented in the query as: Male Married Study

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value M will be replaced by Male.

Start: : Starting position in the string where the information is recorded. The first position is always 1.

Size: : Number of bytes that the information occupies in the string.

Multiple values (binary):

It allows you to retrieve several information stored in decimal form of the selected field.

As examples we will use the following list (although not informed in the interface of inclusion of values in the list, each item has a value assigned automatically according to the order):

Assigned value Description in Lookup

1	Sports
2	Culture

Assigned value Description in Lookup

8	Reading
16	Music

To display the data, a decomposition of the decimal number stored in the bank is performed. For example, numbers 11 and 12 (written in the database) would be decomposed as follows:

11 = 1 + 2 + 8 = (Sports - Culture - Reading)

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

12 = 4 + 8 = (Leisure - Reading)

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

Label:

Text that will be displayed in the Grid.

Include Button:

Includes in the list the values populated in the Label and value fields.

Alter Button:

It changes the attributes of the selected item.

Delete Button:

Deletes the selected item.

Clean Button:

It cleans the fields

Save Button:

Allows the user to save the entire list, for later use using the Load Lookup definition option.

Load Lookup definition:

Allows you to load a predefined value list for use as a lookup.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

Display Settings

You can set CSS values for the field individually. Thus, for example, you can highlight one field of the others in the application. As CSS field properties, when changed, they are added to a class created automatically by ScriptCase for each application field.

Individual field CSS settings, when inserted, override theme settings ([CSS of applications \(Themes\)](#)) selected for application.

The settings are divided into three property blocks, these blocks are:

CSS of Title

This makes it possible to change the CSS properties of the field's Label.

In the example below, You can see the difference of field title configurations. While the fields **provider** and **Category** have the same formatting, inherits the theme of the application, the field **product evaluation** have a different formation of of others, from the changes made in the CSS of the field.

CSS of field

Changes the CSS properties of the <td> where the input object (where the user enters data for insertion into a form) is positioned. In the image below, you can see where the change is applied.

Field with changes to Field CSS properties, changing background color and horizontal alignment

CSS of Input object

Changes the CSS properties in the Input of field object, where the user type the data in a form.

Field with changes to CSS properties of Input Object, changing background color and input source color

CSS properties

The available configuration options are basically the same for each of the configuration blocks above.

description of available configuration attributes

Source

Changes the font of the text according to the fonts selected using the *font-family* property.

In this option, some types of fonts are provided to you (as shown below).

Size

Changes the font size of the text using the property *font-size* in the field class.

You need to select the available value from our list, the measure used for this property is the pixel.

Font Color

Define the font color used by the property *color* in the field class.

Background color

Define the background color using the property *background-color* in the field class.

The colors that will be used in the two color properties listed above, **Font color** and **Background color**, can be entered via the color palette - ![Color palette icon][palette] - available next to the field or manually entered values in the accepted formats that are: *Hexadecimal, RGB, RGBA, HSL, HSLA* or *Color name*.

Color palette

By clicking on the color palette icon -![Color palette icon] [palette] - next to the field, a window will open with some default colors.

When you select one of the colors, a value in hexadecimal format (HEX) will be entered, representing the chosen color.

Hexadecimal

Acronym for hexadecimal, this code is composed of the pound sign (#) plus six digits. The first two define the intensity of the color red, the middle two are green and the last two are blue.

This code is represented by #000000.

In this representation system #000000 represents the color black and #fffff represents the color white.

RGB / RGBA Acronym for Red, Green and Blue. This code is represented by RGB (255,255,255).

The first three numbers defined with intensity of red color, the middle three numbers are green and the last three numbers are blue.

HSL / HSLA

Color name

You need to enter the name of the color you want to use. This name must be entered in English and the color must be within the list of browser supported colors you will use.

Example:

Some sites provide a color list with each of the above values.

This property allows you to specify colors in different ways using different ways of value assignment

You can also enter values directly in the field without using our colors palette. This property lets you specify as colors different ways to assign values to CSS properties that support colors.

Bold

Applies bold style to the font.

Underline

Lets you apply the underlined style to the font.

Border style

Defines the border font style.

Border Collapse

Defines the border collapse.

Border size

Changes the size of the title border.

Border color

Chooses the border color , using a color palette to apply to the title.

Horizontal alignment

Position the filter label at desired location (left,rigth,center e justify).

Vertical alignment

Position the label of the filter in the wanted location baseline, sub, super, top, text-top, middle, bottom, text-bottom).

Not available in Input Object CSS

Width

To set the width of the title.

Height

To set the height of the title.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Calendar
4. Decimal

Decimal

General Settings

Decimal field configuration Interface.

- **Data Type** : Select the type of field for the application. When set to Decimal, you can inform decimal values to the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be easier to the user understand if the name is "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Defines the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Defines the amount of characters allowed for the field.
- **Validation Image** : Displays an image next to the field when it's being validated.
- **Save Variable** : Saves a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Defines the name for the session variable that'll receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Applies the Regional Settings to format the number of the field. When not enabled, the attributes Digit Grouping Separator, Decimal Separator, Negative sign and Negative number format.
- **Digit grouping** : Defines if the field will display the digits separator.
- **Maximum Size** : Determines the max size of the field.
- **Minimum Value** : Set the min value of the field.
- **Maximum Value** : Determines the max value of the field.
- **Decimal Precision** : Defines the amount of decimal places for the field.
- **Digit Grouping Separator** : Defines the character that is used to separate the thousand.
- **Decimal Separator** : Defines the decimal separator.
- **Negative sign** : Defines the character that will be used to display negative numbers.
- **Negative number format** : Defines the position of the Negative sign em relation to the value.
- **Accept** : Determines if the field, will accept only negative, positive or both numbers.
- **Show a calculator** : Displays an icon next to the field, which allows the user to calculate a value for the field.
- **Auto-complete with zeros** : If enabled, it automatically informs the decimal places when not specified. If disabled, the user needs to always inform the decimal value.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced by the curly brackets {}. While running the application, the field within the curly brackets is replaced by its value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Calendar
4. Currency

Currency

General Settings

Currency field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Currency, you can currency values to the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through Javascript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the datatype of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the number of the field. When not enabled, the attributes Digit Grouping Separator, Decimal Separator, Negative sign and Negative number format.
- **Digit grouping** : Defines if the field will display the digits separator.
- **Currency symbol usage** : Defines if the field will display the Currency Symbol of the Regional Settings.
- **Maximum Size** : Determines the max size of the field.
- **Minimum Value** : Determines the min value of the field.
- **Maximum Value** : Determines the max value of the field.
- **Decimal Precision** : Defines the amount of decimal places for the field.
- **Digit Grouping Separator** : Defines the character that is used to separate the thousand.
- **Decimal Separator** : Defines the decimal separator.
- **Negative sign** : Defines the character that will be used to display negative numbers.
- **Negative number format** : Defines the position of the Negative sign em relation to the value.
- **Accept** : Determines if the field, will accept only negative, positive or both numbers.
- **Show a calculator** : Displays an icon next to the field, which aids the user to calculate a value for the field.
- **Auto-complete with zeros** : If enabled, it automatically informs the decimal places when not specified. If disabled, the user needs to always inform the decimal value.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGRES and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced by the curly brackets {}. While running the application, the field within the curly brackets is replaced by its value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Calendar
4. Date

Date

General Settings

Date field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Date, you can inform a date.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows it is value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator attribute.
- **Date separator** : Allows you to inform the separator symbol for the date.
- **First Day** : Define the first day of the week.
- **Display** : Select the format of the day for display.
- **Internal Format** : Allows to define the format the field when the SQL type is different from **Date**. You need to use the characters **Y**, **M** and **D** that correspond to **Year**, **Month** and **Day**.
- **Use Combo-box** : Allows you to select the date using a combo-box.
- **Minimum Date** :
 - **Fixed Date** : Inform the minimum date allowed presented in the selected date format.
 - **Current Date** : Clicking on the icon next to the text field, ScriptCase offers the following options:
 - **Actual date** : It will place the current date as the max allowed date.
 - **Actual date with increment** : The minimum date will be the current date (+) the days or months or years that you want to increment.
 - **Actual date with decrement** : The minimum date will be the current date (-) the days or months or years that you want to decrement.
- **Maximum Date** :
 - **Fixed Date** : Inform the minimum date allowed presented in the selected date format.
 - **Current Date** : Clicking on the icon next to the text field, ScriptCase offers the following options:
 - **Actual date** : It will place the current date as the max allowed date.
 - **Actual date with increment** : The minimum date will be the current date (+) the days or months or years that you want to increment.
 - **Actual date with decrement** : The minimum date will be the current date (-) the days or months or years that you want to decrement.
- **Display Format** : Enables the Date format beside the field when informing the date.
- **Display Calendar** : Enables the a calendar icon beside the field, this allows to select the date from a calendar with the format already setup.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGRES and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. [Calendar](#)
- 4. [Time](#)

Time

General Settings

Time field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Time, you can inform a time to this field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator and time separator attributes.
- **Time separator** : Allows you to inform the separator symbol for the time.
- **Display** : Select the format of the time for display.
- **Internal Format** : Allows to define the format the field when the SQL type is different from **TIME**. You need to use the characters **HH**, **II**, and **SS** that correspond to **Day**, **Hour**, **Minutes** and **Seconds**.
- **Display Format** : Enables the Time format beside the field when informing the time.
- **Use Time picker** : Use the JQuery plug-in to choose the time.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets `{}`. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.

- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Calendar
4. Date and Hour

Date and Hour

General Settings

Date and Time field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Datetime, you can inform a date and time to this field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows it is value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator and time separator attributes.
- **Date separator** : Allows you to inform the separator symbol for the date.
- **Time separator** : Allows you to inform the separator symbol for the time.
- **Display** : Select the format of the day/time for display.
- **Internal Format** : Allows to define the format the field when the SQL type is different from **DATETIME**. You need to use the characters **Y, M, D, HH, II, and SS** that correspond to **Year, Month, Day, Hour, Minutes and Seconds**.
- **Display Format** : Enables the Date and Time format beside the field when informing the date.
- **Group Date and Time** : Groups the date and time in the same field.
- **Display Calendar** : Enables the a calendar icon beside the field, this allows to select the date from a calendar with the format already setup.
- **New Calendar** : Show the new jquery calendar, or the old calendar.
- **Years Limit** : The amount of years that will display in the calendar.
- **View week number** : Display the week number in the calendar.
- **Additional months** : Display additional months in the calendar.
- **Show Combo year and month** : Allows you to select the year and month using a combo-box.
- **Decimal of seconds** : Defines the amount of decimals for the seconds.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.** : When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced by the curly brackets {}. While running the application, the field within the curly brackets is replaced by its value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Calendar
- 4. HTML Image

HTML Image

General Settings

HTML Image field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as HTML Image, it allows to place an image to the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Image** : Defines an image that will be displayed. The icon "Select Image", lists all the standard images from scriptcase and also the image that you have uploaded to scriptcase. The icon "Upload an image" allows the developer to send an image to the server which is from another machine.
- **Border** : Border size for the image in Pixels.
- **Width** : Define the width of the image in Pixels.
- **Height** : Define the height of the image in Pixels.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Calendar
4. Label

Label

General Settings

Label field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as label, the field only display the output of the value .
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Text** : You can inform the text that will display beside the field.
- **Reload** : When enabled, make the text informed for the field to become a link to reload the form when clicked.
- **Position** : Allows to position the field in the chosen block.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. [Calendar](#)
4. [Card](#)

Card

General Settings

Credit Card Number Configuration Interface.

- **Data Type** : Defines the type of field for the application. When it is defined as a Credit Card Number, the field verifies if the value is valid.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name would be "Client Name".
- **Watermark** : Creates a placeholder on the field with the text informed.
- **Initial Value (type)** : Set the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Defines the amount of characters allowed for the field.
- **Validation Image** : Display an image next to the field when it is being validated.
- **Save Variable** : Saves a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Set the name for the session variable that will receive the field value.
- **Field size in database** : Determines the size in bytes of the fields. It is used to determine the max amount of characters to be typed in.
- **Field Mask** : Defines the display mask for the field. There are three types of masks that can be merged.

Character	Description
9	Represents a numeric character (0-9)
A	Represents an alpha numeric character (A-Z,a-z)
*	Represents any alpha numeric character (A-Z,a-z,0-9) typed by the user.

It is possible to merge two or more masks simultaneously, separated by a semi coma with the smallest mask at start. The replacement occurs when the user is typing when the lowest amount of character exceeds.

Examples of Masks

Field	Mask	Informed Value	Formatted Value
Telephone Number	+99 99 9999 - 9999	123456789012	+12 34 5678 - 9012
Software Key (Only Numbers)	AAAA-AAAA-AAAA-AAAA	QWERTYUIASDFGHJK	QWER-TYUI-ASDF-GHJK
Software Key (Letters an Numbers)	**_**_**	Q1W2E3R4T5Y6U7I8	Q1W2-E3R4-T5Y6-U7I8
License Plates	AAA - 9999	QWE1234	QWE - 1234
ScriptCase License	A999A999A999.**	D111H111G111DG2P	D111H111G111-DG2P
Multiple Masks (Telephone)	9999-9999;(99)9999-9999; 9999 999 9999		+99 99 9999-9999

- **Hidden Field** : This option makes the field hidden, but still allows it is value to be processed through JavaScript or PHP.
- **Label Field** : This option make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Display the data type of field in the database.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGRES and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Calendar
- 4. Card type

Card type

General Settings

Credit Card field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Credit Card , it'll setup a combo-box for you to select a creditcard type.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it'll be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that'll receive the field value.
- **Field size in database** : Determines the field size in bytes. It's used to determine the max size of characters that's allowed to type in.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that'll be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface.

- **Credit Card** : Allows to define which cards will be listed and their internal value.
- **Use Title/Deny** : Allows to inform a new value if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen. The label and the value of this option are defined as the Negative Value and the Title.
- **Negative value** : Value of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Title** : Label of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Related Field** : You'll inform which of the fields contains the values of a Credit Card Number so that the validations can be done correctly.
- **Reload Form** : Reloads the form when the field's value is modified.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGRES and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Date of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookups Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Calendar
4. E-mail

E-mail

General Settings

Email field configuration Interface.

- **Data Type** : Defines the type of field for the application. When set to Email, the field applies validations for an email format.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be easier to the user understand if the name is "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Defines the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Defines the amount of characters allowed for the field.
- **Validation Image** : Displays an image next to the field when it is being validated.
- **Save Variable** : Saves a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Defines the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through Javascript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Display Icon** : Displays the Email icon next to the field.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface.

- **Credit Card** : Allows to define which cards will be listed and their internal value.
- **Use Title/Deny** : Allows to inform a new value if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen. The label and the value of this option are defined as the Negative Value and the Title.
- **Negative value** : Value of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Title** : Label of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Related Field** : You'll inform which of the fields contains the values of a Credit Card Number so that the validations can be done correctly.
- **Reload Form** : Reloads the form when the field's value is modified.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.**: When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.

- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Calendar
4. URL

URL

General Settings

URL field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to URL, you can inform a Link value to the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Validation Image** : Allows to display an image next to the field when it is being validated.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Display Icon** : Displays an icon beside the field.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **Show the URL as a clickable link** : Sets the content of the field to clickable link.
- **Target handling where the link will open** : Target where the the lick will direct to after being clicked
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface.

- **Credit Card** : Allows to define which cards will be listed and their internal value.
- **Use Title/Deny** : Allows to inform a new value if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen. The label and the value of this option are defined as the Negative Value and the Title.
- **Negative value** : Value of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Title** : Label of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Related Field** : You'll inform which of the fields contains the values of a Credit Card Number so that the validations can be done correctly.
- **Reload Form** : Reloads the form when the field's value is modified.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is be accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.** : When enabled, the lookup settings is enabled and opens more settings.

- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets `{}`. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.

- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Calendar
4. HTML Color

HTML Color

General Settings

HTML Color field configuration Interface.

- **Data Type** : Select the type of field for the application. When set to HTML Color, you can select a color to be used in the form.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Initial Value (type)** : Set the initial value to the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Defines the amount of characters allowed for the field.
- **Validation Image** : Displays an image next to the field when it is being validated.
- **Save Variable** : Save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Defines the name for the session variable that will receive the field value.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of the field in the database.

Values Format

Format of Values Interface.

- **Credit Card** : Allows to define which cards will be listed and their internal value.
- **Use Title/Deny** : Allows to inform a new value if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen. The label and the value of this option are defined as the Negative Value and the Title.
- **Negative value** : Value of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Title** : Label of the new option if the one of the previous options (American Express, Diners, MasterCard and Visa) wasn't chosen.
- **Related Field** : You'll inform which of the fields contains the values of a Credit Card Number so that the validations can be done correctly.
- **Reload Form** : Reloads the form when the field's value is modified.

Database Value

Allows to define a value to be saved to the database.

Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGRES and Firebird, it is necessary to inform the name of the sequence.

Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Lookup Settings

Method used to place a description next to the field.

- **Use lookup to display the field description.** : When enabled, the lookup settings is enabled and opens more settings.
- **SQL Command** : Defines the SQL command that is going to recover the vales from the database. To build a SQL command, it is possible to use the SQL Builder tool. The SQL format needs to have the following format:

```
SELECT Field_displayed FROM table WHERE Key_Field = '{Form_Field}'
```

The Form field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Font** : Defines the font used to display the lookup.
- **Font Color** : Defines the color in HEX for the lookup display.
- **Font Size** : Defines the font's size in the lookup display.
- **EOF Message** : Defines a message that will be displayed if there is not any results for the record.
- **Use in validation** : When enabled, it will generate an automatic validation based on the lookup, in case there is not any records found, the message will display as a warning (when Updating or Including a record).

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:


Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Calendar
4. HTML Editor


HTML Editor

General Settings

 HTML Editor field configuration Interface.

- **Data Type** : Defines the type of field for the application. When set to HTML Editor, you can inform any type of character and they will be saved in HTML form.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Height** : Set the Height in pixels to the HTML Editor field.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through Javascript or PHP.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **SQL Type** : Displays the data type of field in the database.

Toolbar

 HTML Editor toolbar settings Interface.


- **Properties**
 - **Position** : Location of the HTML Editor toolbar.
 - **Alignment** : Toolbar button alignment.
 - **Status** : Status Bar Display (Do not Display, Top and Bottom).
 - **Amount** : The Amount of lines of the HTML Editor toolbar.
- **Button Organization** : Positions the toolbar buttons.
- **Preview** : You can visualize the toolbar according to the previous settings.

Database Value


Allows to define a value to be saved to the database.

 Database Values configuration Interface.

- **Defined Value** : Allows to specify the fixed value associated to global variables.

 Defined Value configuration Interface.

- **Auto Increment (automatic)** : Allows to use the increment generated by the database. Used only when the field of the database is the type AUTO INCREMENT or similar. For the databases that use sequence like Oracle, PostGres and Firebird, it is necessary to inform the name of the sequence.

 Auto Increment configuration Interface.

- **Auto Increment (manual)** : The application generated simulates an auto increment on the field. To include a value to the field it will automatically calculate the next id.
- **Date of Insertion** : When inserting a record, the field will contain the server date as a value.
- **Data of Update** : When updating a record, the value of the field will be the date of the server.
- **User IP** : The field will receive the IP of the machine that the application is accessed by.
- **Calculated by the database** : The value of the field will be applied by the database. The field will not be used in update processes. For example: it will be updated by a Trigger.
- **Null** : The field will be informed the value null.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** : Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** : Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** : Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.

- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the Input Object.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
- **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the Input Object.
- **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Calendar
- 4. Localization

Localization

General Settings

Location field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it's defined as Location, you will see a combobox with the Languages that are part of the Project Properties.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Reload** : When enabled, make the text informed for the field to become a link to reload the form when clicked.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Calendar
- 4. Theme

Theme

General Settings

Theme field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Theme, you view a combo box with the list of themes that are set to your project.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Reload** : Reload the form after modifying the value.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. [Calendar](#)
- 4. [Youtube](#)

Youtube

General Settings

YouTube field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a YouTube, it allows to display a video from youtube on the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Amount of characters** : Define the amount of characters allowed for the field.
- **Display Mode** : Defines how the you video will display in the application.
- **Width** : Width of the video in pixels.
- **Height** : Height of the video in pixels.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

-
- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
 - **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
 - **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS

Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Calendar
- 4. Google Maps

Google Maps

General Settings

Google Maps field Configuration Interface.

- **Data Type** : Select the type of field for the application. When it is defined as Google Maps, it will use the Google Maps API to display the map in the Form Applications
- **Label** : Set the title to the field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Home** : Defines what type of parameters will be used in the API.

Home Configuration Interface.

- **Display Mode** : Indicates the display mode of the map. It can be opened in a Modal or in a new Window.
- **Width** : Defines the width of the map that is going to be displayed.
- **Height** : Defines the height of the map that is going to be displayed.
- **Zoom** : Defines the initial Zoom (available from the Google API) of the Map location.
- **API Key** : API Key for authorization to use Google Maps in the Application. (Required only for the versions 2 or earlier of the Google API.)

The API Key is a unique key, composed by a string(text) alphanumeric, which is the license to use the service. When you subscribe to use the service, the key is tied to the domain and the directory of the server. All the pages that use the API needs to be in the same directory that was used for the subscription. In case you have a web server on your local machine, you just need to possess a key for testing, and to do this you only need to set http://localhost in the domain of the subscription.

To get your API Key access the site by [clicking here](#)

- **Link Type** : Defines how the link will be displayed.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Calendar
4. Image (Database)

Image (Database)

General Settings

Image (Database) field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as an Image (Database), all the Images files are stored and loaded directly from the Database.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Image Border** : Width of the Image border. Value in Pixels.
- **Image Height** : Image height size. Value in Pixels.
- **Image Width** : Image width size. Value in Pixels.
- **Progress bar** : Displays a progress bar when sending the files to the server.
- **Upload area** : Displays a drag and drop area to upload the file.
- **Maintain Aspect** : Maintains the original aspect ratio of the image when resizing it.
- **Display link only** : When enabled, you will be presented with a link to open the image in another image with the original size of the image (without using the re-dimensioning).
- **Open in Another Window** : Allows to open the image in another window.
- **Extensions to allow** : Extensions allowed to upload, separated by a semi coma (;). If left in blank, all the extensions are allowed. e.g. `docx;jpg;png;xls`
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Multi-upload

This option allows to upload multiple files and store them on a table referencing a record from another table.

To use this option you first need to create a virtual field.

Multi-Upload Configuration.

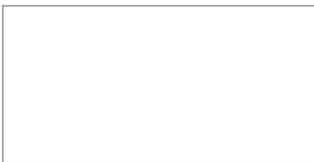
- **Amount of columns** : Amount of columns to displays the files in the Field.
- **Files sorting** : Enables files sorting after the upload.
- **Position of the delete** : Checkbox option to delete files position.
- **Position on the upload** : Position of the records when uploading.
- **Show status** : Display the upload status.
- **Table** : Select the table that will store the information of the files when uploading.

After selecting the table, you will see a list of all the fields of the chosen table, where you will associate the parameters for each field when inserting a new record or updating an existing record.

Multi-Upload fields of the table Configuration.

- **File Name** : Obtains the name of the file that is being loaded.
- **Upload** : Obtains the binary value of the file.
- **Auto Increment** : Used only when field in the database in an auto increment or similar.
- **Manual Increment** : The form will be responsible for managing the auto-increment, calculating the value of the field before inserting the record.
- **Foreign Key** : Associates the file with a value of a field in the current form.
- **Defined** : You can specify a constant value or use a global variable. `[global_var]`
- **Date of Insertion** : Obtains the data of the server when inserting the record.
- **Datetime of Insertion** : Obtains the data and time of the server when inserting the record.
- **User IP** : The field will contain the IP of the computer that is accessing the application.
- **Calculated by the Database** : The value of the field is applied in the database.
- **Calculated by the Database if empty** : The value of the field is applied in the database if not informed by the application.
- **Null** : The field will the null value.

Watch below a video about Multiupload option:



Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.

- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Calendar
4. Image (File Name)

Image (File Name)

General Settings

Image (File Name) field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as an Image (File Name), all the images files are stored and loaded in a directory of the server (Only the image name is stored in the database).
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Image Border** : Width of the Image border. Value in Pixels.
- **Image Height** : Image height size. Value in Pixels.
- **Image Width** : Image width size. Value in Pixels.
- **Maintain Aspect** : Maintains the original aspect ratio of the image when resizing it.
- **Progress bar** : Displays a progress bar when sending the files to the server.
- **Upload area** : Displays a drag and drop area to upload the file.
- **Increment file** : Increments the file name if the theirs an existing one with the same name in the upload folder.
- **Sub-folder** : Sub-folder's name that the files are stored. The is sub folder is relative to the directory defined for the Document upload (see the Settings). It is possible to use global variables or local variables to format the name of the sub-folder.
- **Create Sub-folder** : When enabled, it creates the sub-folder if not already created.
- **Image Caching** : Time in minutes that the image cache will be stored.
- **Hide image name** : When enabled, only the image will be displayed without it is name.
- **Files Deletion** : Deletes the files from the directory when the record is deleted from the database.
- **Display link only** : WWhen enabled, you will be presented with a link to open the image in another image with the original size of the image (without using the re-dimensioning).
- **Open in Another Window** : Allows to open the image in another window.
- **Extensions to allow** : Extensions allowed to upload, separated by a semi coma (;). If left in blank, all the extensions are allowed. e.g. docx;jpg/png;xls
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Multi-upload

This option allows to upload multiple files and store them on a table referencing a record from another table.

To use this option you first need to create a virtual field.

Multi-Upload Configuration.

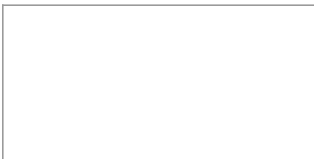
- **Amount of columns** : Amount of columns to displays the files in the Field.
- **Files sorting** : Enables files sorting after the upload.
- **Position of the delete** : Checkbox option to delete files position.
- **Position on the upload** : Position of the records when uploading.
- **Show status** : Display the upload status.
- **Table** : Select the table that will store the information of the files when uploading.

After selecting the table, you will see a list of all the fields of the chosen table, where you will associate the parameters for each field when inserting a new record or updating an existing record.

Multi-Upload fields of the table Configuration.

- **File Name** : Obtains the name of the file that is being loaded.
- **Upload** : Obtains the binary value of the file.
- **Auto Increment** : Used only when field in the database in an auto increment or similar.
- **Manual Increment** : The form will be responsible for managing the auto-increment, calculating the value of the field before inserting the record.
- **Foreign Key** : Associates the file with a value of a field in the current form.
- **Defined** : You can specify a constant value or use a global variable. [global_var]
- **Date of Insertion** : Obtains the data of the server when inserting the record.
- **Datetime of Insertion** : Obtains the data and time of the server when inserting the record.
- **User IP** : The field will contain the IP of the computer that is accessing the application.
- **Calculated by the Database** : The value of the field is applied in the database.
- **Calculated by the Database if empty** : The value of the field is applied in the database if not informed by the application.
- **Null** : The field will the null value.

Watch below a video about Multiupload option:



Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** : Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.

- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
- **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Calendar
4. Document (Database)

Document (Database)

General Settings

Document (Database) field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Document (Database), all the document files are stored and loaded directly from the Database.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Icon** : Displays an icon beside the field to identify the type of document.
- **Progress bar** : Displays a progress bar when sending the files to the server.
- **Upload area** : Displays a drag and drop area to upload the file.
- **File Name** : Defines the field that contains the name of the document stored in the database (the field that contains this information also needs to be stored in the database).
- **File Size** : Defines the field that contains the file size.
- **Extensions to allow** : Extensions allowed to upload, separated by a semi coma (;). If left in blank, all the extensions are allowed. e.g. docx;jpg;png;xls
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Multi-upload

This option allows to upload multiple files and store them on a table referencing a record from another table.

To use this option you first need to create a virtual field.

Multi-Upload Configuration.

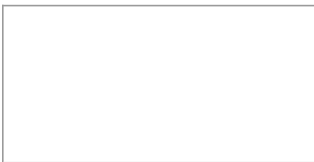
- **Amount of columns** : Amount of columns to displays the files in the Field.
- **Files sorting** : Enables files sorting after the upload.
- **Position of the delete** : Checkbox option to delete files position.
- **Position on the upload** : Position of the records when uploading.
- **Show status** : Display the upload status.
- **Table** : Select the table that will store the information of the files when uploading.

After selecting the table, you will see a list of all the fields of the chosen table, where you will associate the parameters for each field when inserting a new record or updating an existing record.

Multi-Upload fields of the table Configuration.

- **File Name** : Obtains the name of the file that is being loaded.
- **Upload** : Obtains the binary value of the file.
- **Auto Increment** : Used only when field in the database in an auto increment or similar.
- **Manual Increment** : The form will be responsible for managing the auto-increment, calculating the value of the field before inserting the record.
- **Foreign Key** : Associates the file with a value of a field in the current form.
- **Defined** : You can specify a constant value or use a global variable. [global_var]
- **Date of Insertion** : Obtains the data of the server when inserting the record.
- **Datetime of Insertion** : Obtains the data and time of the server when inserting the record.
- **User IP** : The field will contain the IP of the computer that is accessing the application.
- **Calculated by the Database** : The value of the field is applied in the database.
- **Calculated by the Database if empty** : The value of the field is applied in the database if not informed by the application.
- **Null** : The field will the null value.

Watch bellow a video about Multiupload option:



Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.

- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
- **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the field.
- **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Calendar
4. Document (File Name)

Document (File Name)

General Settings

This option allows to upload multiple files and store them on a table referencing a record from another table.

To use this option you first need to create a virtual field.

Multi-Upload Configuration.

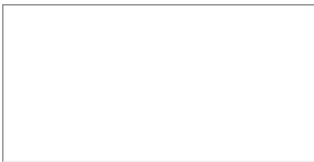
- **Amount of columns** : Amount of columns to displays the files in the Field.
- **Files sorting** : Enables files sorting after the upload.
- **Position of the delete** : Checkbox option to delete files position.
- **Position on the upload** : Position of the records when uploading.
- **Show status** : Display the upload status.
- **Table** : Select the table that will store the information of the files when uploading.

After selecting the table, you will see a list of all the fields of the chosen table, where you will associate the parameters for each field when inserting a new record or updating an existing record.

Multi-Upload fields of the table Configuration.

- **File Name** : Obtains the name of the file that is being loaded.
- **Upload** : Obtains the binary value of the file.
- **Auto Increment** : Used only when field in the database in an auto increment or similar.
- **Manual Increment** : The form will be responsible for managing the auto-increment, calculating the value of the field before inserting the record.
- **Foreign Key** : Associates the file with a value of a field in the current form.
- **Defined** : You can specify a constant value or use a global variable. [global_var]
- **Date of Insertion** : Obtains the data of the server when inserting the record.
- **Datetime of Insertion** : Obtains the data and time of the server when inserting the record.
- **User IP** : The field will contain the IP of the computer that is accessing the application.
- **Calculated by the Database** : The value of the field is applied in the database.
- **Calculated by the Database if empty** : The value of the field is applied in the database if not informed by the application.
- **Null** : The field will the null value.

Watch bellow a video about Multiupload option:



Multi-upload

This option allows to upload multiple files and store them on a table referencing a record from another table.

To use this option you first need to create a virtual field.

Multi-Upload Configuration.

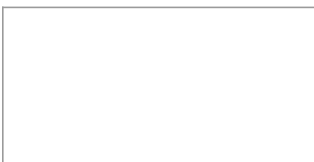
- **Amount of columns** : Amount of columns to displays the files in the Field.
- **Files sorting** : Enables files sorting after the upload.
- **Position of the delete** : Checkbox option to delete files position.
- **Position on the upload** : Position of the records when uploading.
- **Show status** : Display the upload status.
- **Table** : Select the table that will store the information of the files when uploading.

After selecting the table, you will see a list of all the fields of the chosen table, where you will associate the parameters for each field when inserting a new record or updating an existing record.

Multi-Upload fields of the table Configuration.

- **File Name** : Obtains the name of the file that is being loaded.
- **Upload** : Obtains the binary value of the file.
- **Auto Increment** : Used only when field in the database in an auto increment or similar.
- **Manual Increment** : The form will be responsible for managing the auto-increment, calculating the value of the field before inserting the record.
- **Foreign Key** : Associates the file with a value of a field in the current form.
- **Defined** : You can specify a constant value or use a global variable. [global_var]
- **Date of Insertion** : Obtains the data of the server when inserting the record.
- **Datetime of Insertion** : Obtains the data and time of the server when inserting the record.
- **User IP** : The field will contain the IP of the computer that is accessing the application.
- **Calculated by the Database** : The value of the field is applied in the database.
- **Calculated by the Database if empty** : The value of the field is applied in the database if not informed by the application.
- **Null** : The field will the null value.

Watch bellow a video about Multiupload option:



Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Calendar
- 4. Barcode

Barcode

General Settings

Configuration Interface of the Barcode Field.

- **Data Type** : DataType of the field for the application.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name would be "Client Name".
- **Disabled Field** : Defines if the field will be disabled in the "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **SQL Type** : Database field type.

Values Format

Configuration Interface of the Barcode Field.

- **Type** : Type of Barcode.
- **Text** : Barcode Text for illustration purposes.
- There are **18 types of barcodes** that are listed below:

Barcode configuration interface.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Calendar
- 4. Select

Select

General Settings

Select field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Select, you can select multiple option from a combo box (Select Field).
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.
- **Use Select2** : Uses the new component for data selection, allowing searches within the select.
- **Display Select2 search area** : Sets whether to display the search field within Select2.

Initial Value Configuration Interface.

- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Field size in database** : Determines the field size in bytes. It is used to determine the max size of characters that is allowed to type in.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through Javascript or PHP.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

• Lookup Methods

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

Selecting the lookup type.

◦ Lookup Method - Automatic

Lookup used to list the values that will be displayed on the Checkbox field. These values will be recovered dynamically using the Select command.

Automatic Lookup Interface.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimitation.

- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter than (;).
- **Height** : Defines the height for the select object.
- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

- **Link** : Allows to create a link to another form allowing to manipulate the list displayed on the select field. After the manipulation, the select object it updated automatically.

◦ Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).

- **Single Value** :

Setting up Single Value Lookup

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Default** : With this option enabled, the selected item will be inserted to the table when committed.
- **Height** : Defines the height for the select object.

- **Multiple Values (delimiter)**

You can store various values for the select field. The values are separated by the informed delimiter. For example: the combination **Sport, Cinema and Tourism** selected in form will be stored like **E;C;T** in case the delimiter is ; (semi coma).

Setting up Multiple Values (delimiter)

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Height** : Defines the height for the select object.

- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man, Single and Read**, in the database would be stored the following value **MSRD**.

Setting up Multiple Values (position)

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Negative** : Value to be saved in the table in case there is not any valued selected on the field.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.
- **Height** : Defines the height for the select object.

- **Multiple Values (binary)**

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute Value	Lookup Description
1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports, Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) =13

Editing Lookup Configuration Multiple Values (Binary)

- **Label** : Text that will be displayed in the item list of the select field.
- **Height** : Defines the height for the select object.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

Displaying the original lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Lookup Method - Actual value

This lookup is used to list all the values in the selected field.

This lookup will apply a "distinct" to your SQL query.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Calendar
4. Double Select

Double Select

General Settings

Double Select field configuration Interface.

- **Data Type** : Select the type of field for the application. When set to Double Select you can have multiple options selected.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Initial Value (type)** : Defines the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Save Variable** : Saves a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Set the name for the session variable that will receive the field value.
- **Field size in database** : Determines the size in bytes of the fields. It is used to set the max amount of characters to be typed in.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through Javascript or PHP.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Displays the data type of field in the database.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or F can be presented like Male or Female or in a dynamic way, getting these values from the database.

Lookup Settings Display for the field. Automatic Lookup Interface..

- **SQL Select Statement** : Defines the SQL command that will get the values displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.

- **Height** : Set the height(lines) of the field interface.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter than (;).
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Selects another specific connection existing in the project. The select command will be done on the second connection.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).

- **Width** : To define a width for the field.
- **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Calendar
- 4. CheckBox

CheckBox

General Settings

CheckBox field configuration Interface.

- **Data Type** : Select the type of field for the application. When set to CheckBox, you can have multiple options selected.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Initial Value (type)** : Defines the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Save Variable** : Saves a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Define the name for the session variable that will receive the field value.
- **Field size in database** : Determines the size in bytes of the fields. It is used to determine the max amount of characters to be typed in.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through JavaScript or PHP. .
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or F can be presented like Male or Female or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

• Lookup Methods

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

Selecting the lookup type.

◦ Lookup Method - Automatic

Lookup used to list the values that will be displayed on the CheckBox field. These values will be recovered dynamically using the Select command.

Automatic Lookup Interface.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.

- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Option check all** : Displays two options on the field to check and uncheck all.

- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Selects another specific connection existing in the project. The select command will be done on the second connection.

◦ Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

Manual Lookup Interface.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
 - **Single Value** :

Setting up Single Value Lookup

- **Label** : Text that will be displayed in the item list of the checkBox.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Default** : With this option enabled, the selected item will be inserted to the table when committed.
- **Columns** : Set the amount of columns, for the list of items.

- **Multiple Values (delimiter)**

You can store various values for the checkBox field. The values are separated by the informed delimiter. For example: the combination **Sport, Cinema** and

Tourism selected in form will be stored like **E;C;T** in case the delimiter is ; (semi coma).

Setting up Multiple Values (delimiter)

- **Label** : Text that will be displayed in the item list of the checkBox.
 - **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
 - **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
 - **Columns** : Set amount of columns, for the list of items.
- **Multiple Values (position)**

Stores a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Martial Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in the database would be stored the following value **MSRD**.

Setting up Multiple Values (position)

- **Label** : Text that will be displayed in the item list of the checkBox.
 - **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
 - **Negative** : Value to be saved in the table in case there is not any valued selected on the field.
 - **Start** : Starting position of the string that is going to be stored. The first position is always 1.
 - **Size** : Amount of bytes that is going to occupy in the string.
 - **Columns** : Set the amount of columns, for the list of items.
- **Multiple Values (binary)**

Stores a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

Setting up Multiple Values (Binary)

- **Label** : Text that will be displayed in the item list of the checkBox.
- **Columns** : Allows you to inform the amount of columns, for the list of items.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Saves all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Refreshes the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Option check all** : Displays two options on the field to check and uncheck all.

- **Display original and lookup value**: Displays the value of the field and the value returned from the lookup.

- **Separated by**: Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

• CSS of the Title

- **Font**: Allows to choose the font type, that will be applied to the application field title.
- **Font Size**: Allows to choose the the font size, that will be applied to the application field title.
- **Font Color**: Allows to choose a color for the font from the color pallet.
- **Background Color**: Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold**: Applies the bold style to the font.
- **Underline**: Applies the underline style to the font.
- **Border Size**: Applies the border size to the title of the field.
- **Border Color**: Allows to choose a color for the border, using a color pallet to apply to the title.
- **Horizontal Alignment**: Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment**: Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width**: To define a width for the title of the field.
- **Height**: To define a height for the title of the field.

• CSS of the Field

- **Font**: Allows to choose the font type, that will be applied to the application field.
- **Font Size**: Allows to choose the the font size, that will be applied to the application field.
- **Font Color**: Allows to choose a color for the font from the color pallet.
- **Background Color**: Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold**: Applies the bold style to the font.
- **Underline**: Applies the underline style to the font.
- **Border Size**: Applies the border size to the field.
- **Border Color**: Allows to choose a color for the border, using a color pallet to apply to the field.
- **Horizontal Alignment**: Allows to position the field in the desired position (left,right,center and justify).
- **Vertical Alignment**: Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width**: To define a width for the field.
- **Height**: To define a height for the field.

• CSS of the Input Object

- **Font**: Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
- **Font Size**: Allows to choose the the font size, that will be applied to the application field.
- **Font Color**: Allows to choose a color for the font from the color pallet.
- **Background Color**: Allows to define the color for the Input Object, the color can be selected from the color pallet.
- **Bold**: Applies the bold style to the font.
- **Underline**: Applies the underline style to the font.
- **Border Size**: Applies the border size to the Input Object.
- **Border Color**: Allows to choose a color for the border, using a color pallet to apply to the Input Object.
- **Horizontal Alignment**: Allows to position the Input Object in the desired position (left,right,center and justify).
- **Vertical Alignment**: Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width**: To define a width for the Input Object.
- **Height**: To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Calendar
- 4. Radio

Radio

General Settings

Radio field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Radio, your allowed to select one of the options listed.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Field size in database** : Determines the size in bytes of the fields. It is used to determine the max amount of characters to be typed in.
- **Hidden Field** : This option makes the field hidden, but still allows it's value to be processed through Javascript or PHP. .
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

• Lookup Methods

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

Selecting the lookup type.

◦ Lookup Method - Automatic

Lookup used to list the values that will be displayed on the Checkbox field. These values will be recovered dynamically using the Select command.

Automatic Lookup Interface.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT key_field , display_field FROM table
```

The value of the key_field will be stored in the table field.

- **Columns** : Allows you to inform the amount of columns, for the list of items.

- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.

- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.

- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

◦ Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

Manual Lookup Interface.

- **Label** : Text that will be displayed in the item list of the radio.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Default** : With this option enabled, the selected item will be inserted to the table when committed.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones

saved by the user.

- **Use title** : Allows to display a line of a title on the select object (Title Attribute) associated to a value informed manually (Title internal value).

- **Title internal value** : Value Saved to the field when the line of the title is selected in the object.
- **Title** : Text displayed on the title of the Select object.
- **Reload form when value has changed** : With this option selected, the form will refresh if there is a change to the selected object in the field.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

Displaying the original lookup.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Ajax Settings

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

• CSS of the Title

- **Font** : Allows to choose the font type, that will be applied to the application field title.
- **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
- **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the title of the field.
- **Height** : To define a height for the title of the field.

• CSS of the Field

- **Font** : Allows to choose the font type, that will be applied to the application field.
- **Font Size** : Allows to choose the the font size, that will be applied to the application field.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the field.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
- **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the field.
- **Height** : To define a height for the field.

• CSS of the Input Object

- **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
- **Font Size** : Allows to choose the the font size, that will be applied to the application field.
- **Font Color** :Allows to choose a color for the font from the color pallet.
- **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
- **Bold** : Applies the bold style to the font.
- **Underline** : Applies the underline style to the font.
- **Border Size** : Applies the border size to the Input Object.
- **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
- **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
- **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : To define a width for the Input Object.
- **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Calendar
4. Text Auto-Complete

Text Auto-Complete

General Settings

Text auto complete field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Text auto complete, you can inform a value based on the select statement of the lookup settings and it will manipulate an internal Text for the data.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You'll inform the Defined Value here.
- **Use Select2**: Uses the new component for data selection, allowing searches within the select.
- **Amount of characters for the Select2**: Sets the number of characters to start the search in Select2.
- **Amount of lines for the Select2**: Sets the maximum number of rows to list the search result in Select2.
- **Width for the Select2**: Sets a width for the area for the Select2.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Complete to the Left** : Allows to define the character that will be used to complete the value to the left that the user typed in to the max size of the field defined in the Field size in database option.
- **Field size in database** : Determines the field size in bytes. it is used to determine the max size of characters that is allowed to type in.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Validate on submit** : Validate the field only when the form is submitted.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface

- **Case Settings** : Allows to convert the letters of the field when losing the focus. The options are:
 - **Upper Case** : All in Upper Case.
 - **Lower Case** : All in Lower Case.
 - **Capitalize first word** : Capitalizes the first letter of the word.
 - **Capitalize all words** : Capitalizes the first letter of all the words.
- **Minimum Size** : Allows to define the minimum amount of characters accepted for the field.
- **Maximum Size** : Allows to define the maximum amount of characters accepted for the field.
- **Allowed Characters** : Allows to select a set of characters that can be typed into the field. The options are:
 - **All** : Allows any type of character to be typed into the field
 - **Selected** : Defines a set of characters allowed. Using the configuration below, will be allowed to type letters, numbers and other characters (defined in the attribute More). The letters a, b and c (defined in the Less Attribute) are not allowed.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field.
- **Rows** : Allows to define the amount of records displayed in the field.
- **Width** : Defines the width the size of capture box (Capture Text).
- **Search options** : Allows to define the search settings of the field(Start equal to, Any part and End equals to).
- **Capture Text** : When not enabled, you will only view the field to inform the data. See an example below. *lookup Settings configuration Interface.*
- **Display original and lookup value**: Displays the value of the field and the value returned from the lookup.

lookup Settings validation configuration Interface.

- **Show label with the description** : Displays a label with the description.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

1. [Home](#)
2. [Applications](#)
3. Calendar
4. Number Auto-Complete

Number Auto-Complete

General Settings

Number auto complete field configuration Interface.

- **Data Type** : Define the type of field for the application. When set to Number auto complete, you can inform a value based on the select statement of the lookup settings and it will manipulate an internal number for the data.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Watermark** : Create a placeholder for the selected field.
- **Initial Value (type)** : Allows you to define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
 - **System Date** : The field will receive the current server date.

If you select the System Date, then it is not necessary to inform the Initial Value attribute.

- **Initial Value** : You will inform the Defined Value here.

Initial Value Configuration Interface.

- **Amount of characters** : Define the amount of characters allowed for the field.
- **Save Variable** : Allows to save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Allows to define the name for the session variable that will receive the field value.
- **Complete to the Left** : Allows to define the character that will be used to complete the value to the left that the user typed in to the max size of the field defined in the Field size in database option.
- **Field size in database** : Determines the field size in bytes. It is used to determine the max size of characters that is allowed to type in.
- **Hidden Field** : This option makes the field hidden, but still allows its value to be processed through JavaScript or PHP.
- **Label Field** : This options make the field behave as a Read Only field, not allowing modifications to its value while in Insert or Update Mode.

Label Configuration Interface.

- **Validate on submit** : Validate the field only when the form is submitted.
- **Disabled Field** : Defines if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : HTML Object that will be used to display the field in the form.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values Interface

- **Case Settings** : Allows to convert the letters of the field when losing the focus. The options are:
 - **Upper Case** : All in Upper Case.
 - **Lower Case** : All in Lower Case.
 - **Capitalize first word** : Capitalizes the first letter of the word.
 - **Capitalize all words** : Capitalizes the first letter of all the words.
- **Minimum Size** : Allows to define the minimum amount of characters accepted for the field.
- **Maximum Size** : Allows to define the maximum amount of characters accepted for the field.
- **Allowed Characters** : Allows to select a set of characters that can be typed into the field. The options are:
 - **All** : Allows any type of character to be typed into the field
 - **Selected** : Defines a set of characters allowed. Using the configuration below, will be allowed to type letters, numbers and other characters (defined in the attribute More). The letters a, b and c (defined in the Less Attribute) are not allowed.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Settings Display for the field.

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field.
- **Rows** : Allows to define the amount of records displayed in the field.
- **Width** : Defines the width the size of capture box (Capture Text).
- **Search options** : Allows to define the search settings of the field(Start equal to, Any part and End equals to).
- **Capture Text** : When not enabled, you will only view the field to inform the data. See an example below. lookup Settings configuration Interface.
- **Display original and lookup value** : Displays the value of the field and the value returned from the lookup.

lookup Settings validation configuration Interface.

- **Show label with the description** : Displays a label with the description.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Ajax Processing

Ajax Processing configuration Interface.

Allows to define the field that will be reloaded when selecting a value that has the (onChange) trigger.

For example: a field of the type select with a **list of states**, and another select field with a **list of cities**. When selecting a state, the cities list is reloaded.

Display Settings

The Display Settings allows to define the CSS for the fields individually. For each Display Settings of Scriptcase, exists the same attributes available for this interface.

- **CSS of the Title**
 - **Font** : Allows to choose the font type, that will be applied to the application field title.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field title.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the title.
 - **Horizontal Alignment** : Allows to position the label of the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the label of the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the title of the field.
 - **Height** : To define a height for the title of the field.
- **CSS of the Field**
 - **Font** : Allows to choose the font type, that will be applied to the application field.
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the field, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the field.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the field.
 - **Horizontal Alignment** : Allows to position the field in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the field in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the field.
 - **Height** : To define a height for the field.
- **CSS of the Input Object**
 - **Font** : Allows to choose the font type, that will be applied to the Input Object. For example: Radio, Select, Text, etc
 - **Font Size** : Allows to choose the the font size, that will be applied to the application field.
 - **Font Color** :Allows to choose a color for the font from the color pallet.
 - **Background Color** : Allows to define the color for the Input Object, the color can be selected from the color pallet.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Applies the underline style to the font.
 - **Border Size** : Applies the border size to the Input Object.
 - **Border Color** : Allows to choose a color for the border, using a color pallet to apply to the Input Object.
 - **Horizontal Alignment** : Allows to position the Input Object in the desired position (left,right,center and justify).
 - **Vertical Alignment** : Allows to position the Input Object in the desired position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : To define a width for the Input Object.
 - **Height** : To define a height for the Input Object.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Calendar
- 4. Signature

Signature

General Settings

Configuration Interface of the Signature Field.

The signature field will help you creating more sophisticated forms and making it possible to store signatures in your database. Inside our development environment we have specific settings that will help you to customize your field, those options are:

- **Data Type** : You can define the type of field for the application. When it is defined as a text, it accepts letters, numbers and special characters.
- **Label** : Lets you define a label to the field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Background color** : Defines a color to the field background by using a valid hexadecimal color value or by choosing a color from the color palette.
- **Pen color** : Set a color to the pen by using a valid hexadecimal color value or by choosing a color from the color palette.
- **Width** : Lets you define a width to the field.
- **Height** : Set a height to the field.
- **Subtitle** : Defines the subtitle that will be displayed beside the field.
- **Initial Value** : Lets you define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
- **Save Variable** : Save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Set the name for the session variable that will receive the field value.
- **Disabled Field** : Define if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : Displays the HTML Object that will be used to display the field in the form.
- **SQL Type** : Displays the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of ScriptCase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **CSS of the Title**
 - **Font** : Select the font type, that will be applied to the application field title.
 - **Font Size** : Defines the font size, that will be applied to the application field title.
 - **Font Color** : Choose a color to the font by using a valid hexadecimal color value or from the color picker.
 - **Background Color** : You can define the color for the field by using a valid hexadecimal color value or from the color picker.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Set the underline style to the font.
 - **Border style** : Choose a style for the border.
 - **Collapse** : Defines the collapse for the border.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Choose a color for the border, using a color palette to apply to the title.
 - **Horizontal Alignment** : Position the label of the field in the wanted position (left,right,center and justify).
 - **Vertical Alignment** : Position the label of the field in the wanted position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : Defines a width for the title of the field.
 - **Height** : Set a height for the title of the field.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

□

1. [Home](#)
2. [Applications](#)
3. Calendar
4. Rating

Rating

General Settings

This field allows the developer to create a field for rating using stars (or any other image), where the final user will be able to select a rate option according to the configuration made by the developer.

Description of the attributes available for configuration.

Data type

Define the type of the field in the application. In this case we must select the type **Rating**.

Label

Define the application title that will be showed when the application is executed. The terminology used in the interface of your application is very important for the usability, and we need to use familiar terms for the final user, instead of use terms from the system.

Example, this field "rating" that is named **cmp_rating** however, the customer would have a better understanding if the label uses **Product Rating**.

Besides use a fixed text, the **Label** attribute allows the use of **Langs** to define the field title, allowing the [internationalization of your application](#).

Subtitle

Define the subtitle that will be shown in the field below the ratings. **Example**: "Thank you for your feedback!".

As in the **Label**, the **subtitle** attribute also allows the use of **Langs** for [internationalization of your application](#).

Amount of icons

Define the number of icons that will be displayed in the field. The value set in this attribute must be according to the way the evaluation will be realized.

To evaluate using Stars normally it is used five stars, so , we must to inform 5 in this attribute. After we set the images that will be displayed in the attribute **Enabled icon** and **Disabled icon** the result will be the same as in the example:

To use Like/Dislike **for example** we must inform 1 in the **amount of icons**, so only one image will be displayed and will change according to the click. In this case, the storage in the database will be 1 for like (enabled icon) and 0 for dislike (disabled icon). **You can see the field configured below as like/dislike:**

Enabled icon

Define the icon that will be shown for the amount selected. Following the like/dislike example, in this option, we must insert the Like image . We must do the same for a star evaluation, where for default we must insert the stars filled .

Disabled icon

Define the icon that will be shown for the not selected icons. In this attribute, we set an image to represent the opposite of the previous images set in the **Enabled icon**.

Still referring the example of the Like/Dislike, in this attribute we must set the Dislike .

As in the stars example, we must insert the image with the empty stars .

Initial Value (type)

Allow the definition of an initial value for the field when the form application is in insert mode. The only option available in this field is **Defined**. Selecting this option the attribute **Initial Value** will be showed for set the value.

Example: The initial value is 3 when inserting a new record, the rate field will initiate with 3 stars already selected.

The value defined in this attribute will overlap any value defined previously.

Save variable

Allow saving a session variable ([Global variable](#)) with the field value for being used in other applications.

Example: In the login form, the user name can be saved in the session and used to be displayed in the header of other applications.

Variable Name

In this attribute, we must define a name for the **session variable** that was enabled previously to receive the field value.

We must inform only the variable name, - **var_rating**.

We can recover the value as a [global variable](#).

HTML Type

HTML object used to show the form field.

SQL Type

Inform the type of the field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of ScriptCase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **CSS of the Title**

- **Font** : Select the font type, that will be applied to the application field title.
- **Font Size** : Defines the font size, that will be applied to the application field title.
- **Font Color** : Choose a color to the font by using a valid hexadecimal color value or from the color picker.
- **Background Color** : You can define the color for the field by using a valid hexadecimal color value or from the color picker.
- **Bold** : Applies the bold style to the font.
- **Underline** : Set the underline style to the font.
- **Border style** : Choose a style for the border.
- **Collapse** : Defines the collapse for the border.
- **Border Size** : Applies the border size to the title of the field.
- **Border Color** : Choose a color for the border, using a color palette to apply to the title.
- **Horizontal Alignment** : Position the label of the field in the wanted position (left,right,center and justify).
- **Vertical Alignment** : Position the label of the field in the wanted position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Width** : Defines a width for the title of the field.
- **Height** : Set a height for the title of the field.

Help Settings

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.

Help Description

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

Pages

A page is a container of blocks. Each application has at least one page by default. In Form, Control, and Search applications, there is possible to create many pages. The pages are like Tabs, where each tab contains one or more blocks.

PAGES SETTINGS

ATTRIBUTE	VALUE	DESCRIPTION												
Edit the Page Settings <table border="1"> <thead> <tr> <th>Op</th> <th>Name</th> <th>Title</th> <th>Icon</th> </tr> </thead> <tbody> <tr> <td></td> <td>Pag1</td> <td>Pag1</td> <td></td> </tr> <tr> <td></td> <td>Pag2</td> <td>pag2</td> <td></td> </tr> </tbody> </table> <input type="button" value="Include"/>			Op	Name	Title	Icon		Pag1	Pag1			Pag2	pag2	
Op	Name	Title	Icon											
	Pag1	Pag1												
	Pag2	pag2												
Minimum tab width	<input type="text"/>													
Maximum tab width	<input type="text"/>													
Font	<input type="text"/> Aa													
Font Size	<input type="text"/>													
Selected Font Color	<input type="text"/>													
Selected Background Color	<input type="text"/>													
Non Selected Font Color	<input type="text"/>													
Non Selected Background Color	<input type="text"/>													

Pages (available only in the Form, Control and Search applications) configuration Interface.

See the example below of the Form application using two Pages: General Data and Documents.

EDITING - CUSTOMERS 04/05/2017

Quick search

Pag1 pag2

Customerid * ALFKI

Companyname

Contactname

Contacttitle

Birthdate mm/dd/yyyy

Country

Files

Drag an image here

* Required field(s)

Go to 1 2 3 4 5 [1 of 91]

Form Application using Pages feature.

Pages Settings

The form application already comes with a default page, identified as "Pag1", but it is possible to rename it. Use the pages when you have an application that contains many fields. A form with more than 20 fields in a vertical way is challenging to use. So you could arrange the fields into the blocks and the blocks into the pages.

PAGES SETTINGS														
ATTRIBUTE	VALUE	DESCRIPTION												
<div style="border: 1px solid #ccc; padding: 5px; margin: 5px auto; width: 80%;"> <p style="text-align: center; margin: 0;">Edit the Page Settings</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Op</th> <th>Name</th> <th>Title</th> <th>Icon</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">🗑️</td> <td>Pag1</td> <td>Pag1</td> <td style="text-align: center;"></td> </tr> <tr> <td style="text-align: center;">+</td> <td>Pag2</td> <td>pag2</td> <td style="text-align: center;"></td> </tr> </tbody> </table> <p style="text-align: center; margin: 5px 0;">Include</p> </div>			Op	Name	Title	Icon	🗑️	Pag1	Pag1		+	Pag2	pag2	
Op	Name	Title	Icon											
🗑️	Pag1	Pag1												
+	Pag2	pag2												
Minimum tab width	<input type="text"/>													
Maximum tab width	<input type="text"/>													
Font	<input type="text"/> Aa													
Font Size	<input type="text"/>													
Selected Font Color	<input type="text"/>													
Selected Background Color	<input type="text"/>													
Non Selected Font Color	<input type="text"/>													
Non Selected Background Color	<input type="text"/>													

Pages Configuration Interface.

Including A New Page,

To include a new page, enter the information about the name and the label of the new page, select an image if you wish, then click on the button "Include".

Deleting A Page

To delete a page, click on the Trash icon corresponding to the page line.

Common Settings

Font

Set the font-family of the pages titles. By clicking on the right side icon, you can select the font-family from a list.

Font Size

Set the font size of the pages titles.

Selected Font Color

Set the font color of the selected page.

Selected Background Color

Set the Background Color of the selected page.

Non Selected Font Color

Set the font color of the non selected pages.

Non Selected Background Color

Set the Background Color of the non selected pages.

Blocks

Blocks are "containers" where you can position the application fields of Forms, Controls, or Slide Grids.

Scriptcase creates applications with one block by default. You can add more blocks as you wish, to organize it in the best way.

See below, the Columns Organization, and where you can define the position of the next block: beside or below the current one.

Op	Name	Block	Title			Fields		Organization			
		Label	Display	Display	Position	Columns	Position	Next	Width	Collapse	
Pag1											
+	🗑️	form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▾	1	Beside ▾	Below ▾	100%	Start open ▾
Pag2											
+	🗑️	Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▾	1	Beside ▾	Below ▾	100%	Start open ▾

Application Block configuration

On the left side of each block, there are two icons, the first one to edit the information of the block and the second one to delete the block.

Organizing The Position Of The Blocks

See below how to modify the display order of the Blocks in one Page.

Click and drag the block that you desire to modify to its new position.

Block		Title		Label		Fields			Organization	
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
		form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Pag2										
		Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open

Application Block Display configuration

- see how to remove a block from the display

Click on the block desired and drag it to the item "Blocks not Shown". This way, you can also drag the block to another page if desired. See the images below.

Block		Title		Label		Fields			Organization	
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
		form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Pag2										
		Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Blocks not shown										
		Block 3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open

Application Block Display configuration

Block		Title		Label		Fields			Organization	
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse
Pag1										
		form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Pag2										
		Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open
Blocks not shown										
		Block 3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside	1	Beside	Below	100%	Start open

Application Block Display configuration

Block

- **ame:** The name that identifier the Block.
- **abel:** Title of the block to display in the application.

Title

- **isplay:** It controls the display of the block title.

Label

- **isplay:** It controls the display of the field labels of the block.
- **osition:** Options to display label :
 1. **bove:** Display the label above the field.
 2. **eside:** Display the label beside the field.
 3. **elow:** Display the label below the field.

Fields

- **olumns:** Amount of columns side by side in the block.
- **osition:** The way to display the fields in the block:
 1. **elow:** Display the fields one below the other respecting the number of columns.
 2. **eside:** Display the fields one beside the other respecting the number of columns.
 3. **ine:** Display the fields one beside the other with no tabulation.

Organization

- **ext:** The way to display the blocks in the page:
 1. **elow:** Set to show the following block below the current one.
 2. **eside:** Set to show the following block beside the current one.
 3. **abs:** Set to show the following block in a different tab then the current one.
- **idth:** Set the block width in pixels or percentages. Use the symbol "%" to indicates the value in percentage.
- **ollapse:** Enables the option to close the block.

Create A New Block

To include new blocks in an Application, click on the button . Then, enter the name and label of the block in the following interface and finish by click on Create.

Add New Block

NAME

LABEL

Creating application blocks configuration


Name

Name of the Block.

Label

Title of the block to display in the application.

Edit Blocks

To edit a block, click on the icon  that is on the left side of the block. Then you can see the following interface to define the parameters of the blocks. Click on Save to finish.

ATTRIBUTE	VALUE
Name	<input type="text" value="form_orders"/>
Title	<input type="text" value="form_orders"/>
Display Title	<input type="radio"/> Yes <input checked="" type="radio"/> No
Title Font	<input type="text" value="Aa"/>
Font Size	<input type="text" value=""/>
Font Color	<input type="text" value=""/>
Background Color	<input type="text" value=""/>
Background image	<input type="text" value=""/>
Title Height	<input type="text" value="20"/> pixels
Horizontal Alignment	<input type="text" value=""/>
Vertical Alignment	<input type="text" value=""/>
Display Label	<input checked="" type="radio"/> Yes <input type="radio"/> No
Columns	<input type="text" value="1"/>
Columns Width	<input type="text" value="Calculated"/>
Label Color	<input type="text" value=""/>
Fields Organization	<input type="text" value="Beside"/>
Label Position	<input type="text" value="Beside"/>
Next Block	<input type="text" value="Below"/>
Border Color	<input type="text" value=""/>
Border Width	<input type="text" value="0"/> pixels
Block Width	<input type="text" value="100%"/>
Block Height	<input type="text" value=""/>
Cell Spacing	<input type="text" value=""/> pixels
Collapse	<input type="text" value="Start open"/>

Application Block editing interface

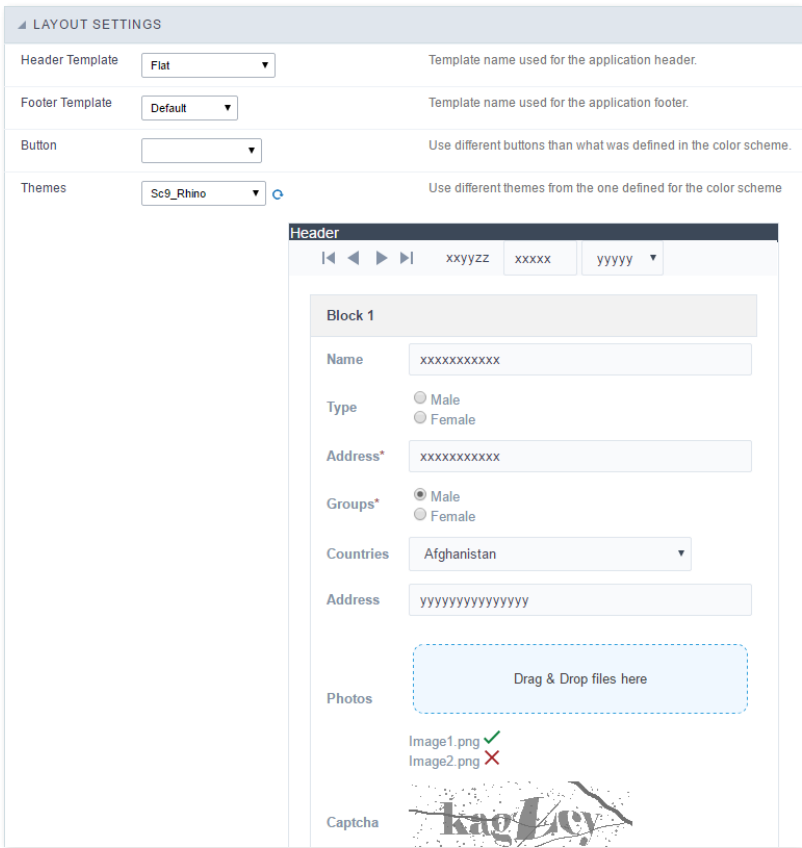
Name

Name of the block. ##### Title
 Block title for display. ##### Display Title
 This option, when active, allows displaying the block title. ##### Title Font
 Set the font family of the block title. ##### Font Size
 Set the font size of the block title. ##### Font Color
 Set the font color of the block title. ##### Background Color
 Set the Background Color of the block title. ##### Background image
 Set a Background image for the block title. ##### Title Height
 Height in pixels of the block title line. ##### Horizontal Alignment
 Horizontal Alignment of the block title (Left, Center, and Right). ##### Vertical Alignment
 Vertical Alignment of the block title (Top, Middle, and Bottom). ##### Display Label
 Display the labels of the fields in the block. ##### Columns
 Amount of field columns in a block. ##### Columns Width
 Set the field column width of the block. ##### Label Color
 Color of the field labels. ##### Fields Organization
 The way to display the fields in the block. ##### Label Position
 Set the position of the field labels of the block. ##### Next Block
 Set the position of the following block relating to the current one. ##### Border Color
 The border Color for the block. ##### Border Width
 The border Width for the block. ##### Block Width
 The width for the block. ##### Block Height
 The Height for the block. ##### Cell Spacing
 The Cell Spacing in the block. ##### Collapse
 It enables the option to close the block.

This module lets you editing display options for the application. You can apply visualization themes, organize blocks, define values, and the display format of the Header and Footer.

Layout Settings

ScriptCase creates an app by using default values defined per project, but you can change those values for a specific application on this interface.



Application Layout Settings

Header Template

Allows choosing the look of the application Header.

Footer Template

Allows choosing the look of the application Footer.

Button

Allows choosing the button theme for the application.

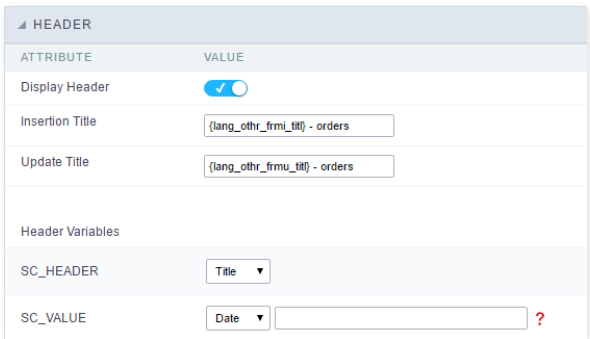
Themes

Choose one of the selected themes in the project properties. It defines the look of the application, like colors, fonts, and others.

Header & Footer

Header

Here we can define if we want to display the header or not, the application title, and fill the header variables. The header looks according to the selected template in the Layout settings interface.



Application Header Configuration

Display Header


This option determines if we want to display the header or not.

Title

Allows informing the title of the application.

Header Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **itle**: It displays the value of "**Application Title**" in the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Depending on the Application, you may have more than one title option.

Footer

FOOTER	
ATTRIBUTE	VALUE
Display Footer	<input type="checkbox"/>
Footer Variables	
NM_FOOTER_DATA	<input type="text"/>
NM_FOOTER_LOGO	<input type="text"/>
NM_FOOTER_TITLE	<input type="text"/>

Application Footer Settings


This page may change depending on the footer template chosen in the Layout Settings.

Display Footer

This option determines if we want to display the footer or not.

Footer Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

In the events you can use global variables, local variables, PHP, JavaScript, HTML, CSS codes and macros.

onApplicationInit

This event occurs only once, when the application is loaded and before the application executes the SQL. It can be used for data processing or checking variables.

Example: `if ([glo_var_dept] != 'financial'){ sc_redirect(app_x.php); }`

onNavigate

This event occurs when navigating between the application pages.

OnScriptInit

This event occurs when the application scripts are started. It runs before the OnLoad events(when present), however, it will always be executed when running, reloading or navigating the application.

onLoadRecord

This event is executed at the moment the record line is loading. The OnRecord event is a loop that loads the records by each row.

onLoad

This event runs when the application is finished loading.

onRefresh

This event runs when the refresh option of the application is enabled.

onValidate

This event runs when validating the information of the fields, when submitting the form.

onValidateFailure

This event runs when the form validation gives an error.

onValidateSuccess

This event runs when the form validation does not give an error.

onBeforeInsert

This event is executed before inserting a record from the application.

onAfterInsert

This event is executed after inserting a record from the application.

onBeforeInsertAll

This event runs before inserting all the selected records from the Multiple Records, Editable Grid and Editable Grid (View) Form.

onAfterInsertAll

This event runs after inserting all the selected records from the Multiple Records, Editable Grid and Editable Grid (View) Form.

onBeforeUpdate

This event is executed before updating a record from the application.

onAfterUpdate

This event is executed after updating a record from the application.

onBeforeUpdateAll

This event runs before updating all the selected records from the Multiple Records, Editable Grid and Editable Grid (View) Form.

onAfterUpdateAll

This event runs after updating all the selected records from the Multiple Records, Editable Grid and Editable Grid (View) Form.

onBeforeDelete

This event runs before deleting the records from the Form.

onAfterDelete

This event runs after deleting the records from the Form.

onBeforeDeleteAll

This event runs before deleting all the selected records from the Multiple Records, Editable Grid and Editable Grid (View) Form.

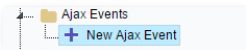
onAfterDeleteAll

This event runs after deleting all the selected records from the Multiple Records, Editable Grid and Editable Grid (View) Form.

OnClick

The ajax event OnClick is executed when the field that it's based on is clicked.

- reating a new ajax event



Creating a new ajax event

- electing a field

Creating Ajax Events	
Select the field to create an event	contacttitle
Select event for Ajax processing	onClick
Fields that will be passed as parameters Double click to check or uncheck	<ul style="list-style-type: none"> customerid companyname contactname contacttitle birthdate country regionid stateid city address postalcode phone
<input type="button" value="Create Event"/>	

Selecting a field of the ajax event

Choose a field to create an event To define in which field the event will be add to.

Choose an event that run the ajax Defines which event will be added to the field.

OnChange

The ajax event OnChange is executed when the value of the field that it's based on is modified.

- reating a new ajax event



Creating a new ajax event

- electing a field

Creating Ajax Events	
Select the field to create an event	contacttitle
Select event for Ajax processing	onClick
Fields that will be passed as parameters Double click to check or uncheck	<ul style="list-style-type: none"> customerid companyname contactname contacttitle birthdate country regionid stateid city address postalcode phone
<input type="button" value="Create Event"/>	

Selecting a field of the ajax event

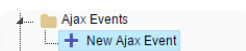
Choose a field to create an event To define in which field the event will be add to.

Choose an event that run the ajax Defines which event will be added to the field.

OnBlur

The ajax event OnBlur is executed when the focus is removed from the field that event is based on.

- reating a new ajax event



Creating a new ajax event

- electing a field

Creating Ajax Events

Select the field to create an event

Select event for Ajax processing

Fields that will be passed as parameters
Double click to check or uncheck

- customerid
- companyname
- contactname
- contacttitle
- birthdate
- country
- regionid
- stateid
- city
- address
- postalcode
- phone

Create Event

Selecting a field of the ajax event

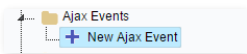
Choose a field to create an event To define in which field the event will be add to.

Choose an event that run the ajax Defines which event will be added to the field.

OnFocus

The ajax event OnFocus is executed when the field that it's based on is applied with a focus.

- reating a new ajax event



Creating a new ajax event

- electing a field

Creating Ajax Events

Select the field to create an event

Select event for Ajax processing

Fields that will be passed as parameters
Double click to check or uncheck

- customerid
- companyname
- contactname
- contacttitle
- birthdate
- country
- regionid
- stateid
- city
- address
- postalcode
- phone

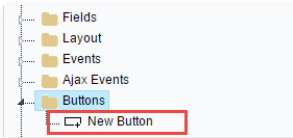
Create Event

Selecting a field of the ajax event

Choose a field to create an event To define in which field the event will be add to.

Choose an event that run the ajax Defines which event will be added to the field.

Including the default applications buttons, you can also create new buttons manually. These buttons can appear in the application toolbar.

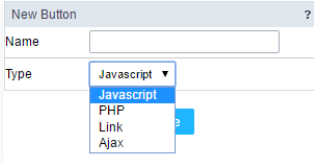


Creating new buttons

Creating a button

To create a new button, click into the option "new button" and type a name and the button type.

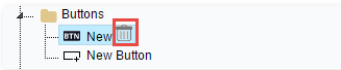
Button types: (JavaScript, PHP, and Ajax).



Button types

Deleting a button

To delete a button click on the icon next to the name of the button in the application menu (recycle bin).



Deleting a button

JavaScript

Display Mode

You can configure the display mode of the javascript button in Image, Button or Link.

Button

Button Settings: JavaScript	
ATTRIBUTE	VALUE
Display Mode	Button ▼
Label	JavaScript
Hint	
Confirmation Message	
Type	JavaScript

Setting up Javascript Button.

- Display Mode** You can select the display mode for the javascript button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Type** Description of the created button.

Image

Button Settings: JavaScript	
ATTRIBUTE	VALUE
Display Mode	Image ▼
Icon	<input type="text"/>
Hint	
Confirmation Message	
Type	JavaScript

Setting up Javascript Button.

- Display Mode** You can select the display mode for the javascript button in this option.
- Icon** Allows you to inform the icon that will be displayed on the button while the execution of the application.

- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Type** Description of the created button.

Link

▶ Button Settings: JavaScript	
ATTRIBUTE	VALUE
Display Mode	Link ▾
Label	JavaScript
Hint	
Confirmation Message	
CSS Style	default ▾
Type	JavaScript

Setting up Javascript Button.

- Display Mode** You can select the display mode for the ajax button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- CSS Style** Name of the CSS class, style created in the layout editor.
- Type** Description of the created button.

Code Block

▶ JavaScript Code	
Theme default ▾	
1	

Javascript button coding block.

In this block, only JavaScript is accepted.

PHP

Display Mode

You can configure the display mode of the PHP button in Image, Button or Link.

Button

▶ Button Settings: PHP	
ATTRIBUTE	VALUE
Display Mode	Button ▾
Label	PHP
Hint	
Confirmation Message	
Reload quantity of records	No ▾
Type	PHP
Target	Same Window ▾

Setting up PHP Button.

- Display Mode** You can select the display mode for the link button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Reload quantity of records** This option is used to update the amount of records in the application.
- Type** Description of the created button.
- Target** Defines the window destination where the code will run (Open in the same Window, Open in another Window, Modal).

Image

ATTRIBUTE	VALUE
Display Mode	Image ▾
Icon	<input type="text"/>
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
Reload quantity of records	No ▾
Type	PHP
Target	Same Window ▾

Setting up PHP Button.

- Display Mode** You can select the display mode for the link button in this option.
- Icon** Allows you to inform the icon that will be displayed on the button while the execution of the application.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Reload quantity of records** This option is used to update the amount of records in the application.
- Type** Description of the created button.
- Target** Defines the window destination where the code will run (Open in the same Window, Open in another Window, Modal).

Link

ATTRIBUTE	VALUE
Display Mode	Link ▾
Label	PHP <input type="text"/>
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
CSS Style	default ▾
Reload quantity of records	No ▾
Type	PHP
Target	Same Window ▾

Setting up PHP Button.

- Display Mode** You can select the display mode for the link button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- CSS Style** Name of the CSS class, style created in the layout editor.
- Reload quantity of records** This option is used to update the amount of records in the application.
- Type** Description of the created button.
- Target** Defines the window destination where the code will run (Open in the same Window, Open in another Window, Modal).

Code Block

ATTRIBUTE	VALUE
Code	*Ajax button coding block. *

In this block, you can use macros, PHP code and JavaScript.

Link

Display Mode

You can configure the display mode of the link button in Image, Button or Link.

Button

ATTRIBUTE	VALUE
Display Mode	Button ▾
Label	Link <input type="text"/>
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
Type	Link

[Link](#)

Setting up Link Button.

Display Mode	You can select the display mode for the link button in this option.
Label	Text of the button that will be display in the application while executing.
Hint	Hint message for the button. (Displayed when the mouse hovers the button).
Confirmation Message	Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
Type	Description of the created button.

Image

ATTRIBUTE	VALUE
Display Mode	Image
Icon	<input type="text"/>
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
Type	Link

Setting up Link Button.

Display Mode	You can select the display mode for the link button in this option.
Icon	Allows you to inform the icon that will be displayed on the button while the execution of the application.
Hint	Hint message for the button. (Displayed when the mouse hovers the button).
Confirmation Message	Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
Type	Description of the created button.

Link

ATTRIBUTE	VALUE
Display Mode	Link
Label	Link
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
CSS Style	default
Type	Link

Setting up link Button.

Display Mode	You can select the display mode for the ajax button in this option.
Label	Text of the button that will be display in the application while executing.
Hint	Hint message for the button. (Displayed when the mouse hovers the button).
Confirmation Message	Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
CSS Style	Name of the CSS class, style created in the layout editor.
Type	Description of the created button.

Setting Up The Link

- electing the applications

Link between applications - Application List

Select the application that will be called:

Application: All By folder By type

- calendar_events
- chart_customers
- dashboard
- form_customers
- form_employees
- form_orders
- form_sec_users
- grid_categories

Choosing the application for the button link.

You should select an application to be called from the button link.

- ink Parameters

Choosing the parameters for the button link.

Field Allows you to use an existing field from the current application as a parameter for the link.

Variable Allows you to use a global variable from the current application as a parameter for the link.

Fixed Allows you to inform a fixed value as a parameter for the link.

Empty No value will be passed as a parameter for the link.

- ink Properties (Grid)

Configuring the properties for the link button when the destined application is a Grid.

Link Operation Mode How the link will open (Open in the same Window, Open in another Window, Modal).

Exit URL for the target application URL or an application that be redirect to when exiting the Grid application.

Initial Mode Allows you to define the initial mode of the grid application (Search or Grid).

Number of Lines Allows you to define the amount of lines displayed in the Grid.

Number of Columns Allows you to define the amount of columns displayed in the Grid.

Paging Enable the paging in the Grid.

Display Header Enable the Grid Header.

Active Navigation Buttons Enable the navigation button (First, Back, Next and Last) in the Grid.

- ink Properties (Form)

Configuring the properties for the link button when the destined application is a Form.

Link Operation Mode How the link will open (Open in the same Window, Open in another Window, Modal).

Exit URL for the target application URL or an application that be redirect to when exiting the Form application.

Enable insert button on target application Enable the "New" button in the Form Application.

Enable update button on target application Enable the "Update" button in the Form Application.

Enable delete button on target application Enable the "Delete" button in the Form Application.

Enable navigation button on target application Enable the navigation button (First, Back, Next and Last) in the Form.

Enable button to edit a grid record Enable the buttons that allow you to edit the records of a Grid

Ajax

Display Mode

You can configure the display mode of the Ajax button in Image, Button or Link.


Button

▶ Button Settings: Ajax	
ATTRIBUTE	VALUE
Display Mode	Button ▾
Label	Ajax
Hint	
Confirmation Message	
Type	Ajax
Show in Update mode	<input checked="" type="radio"/> Yes <input type="radio"/> No
Show in the inserting mode	<input type="radio"/> Yes <input checked="" type="radio"/> No

Setting up Ajax Button.

- Display Mode** You can select the display mode for the Ajax button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Type** Description of the created button.

Image

▶ Button Settings: Ajax	
ATTRIBUTE	VALUE
Display Mode	Image ▾
Icon	<input type="text"/> 
Hint	
Confirmation Message	
Type	Ajax
Show in Update mode	<input checked="" type="radio"/> Yes <input type="radio"/> No
Show in the inserting mode	<input type="radio"/> Yes <input checked="" type="radio"/> No

Setting up Ajax Button.

- Display Mode** You can select the display mode for the Ajax button in this option.
- Icon** Allows you to inform the icon that will be displayed on the button while the execution of the application.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Type** Description of the created button.

Link

▶ Button Settings: Ajax	
ATTRIBUTE	VALUE
Display Mode	Link ▾
Label	Ajax
Hint	
Confirmation Message	
CSS Style	default ▾
Type	Ajax
Show in Update mode	<input checked="" type="radio"/> Yes <input type="radio"/> No
Show in the inserting mode	<input type="radio"/> Yes <input checked="" type="radio"/> No

Setting up Ajax Button.

- Display Mode** You can select the display mode for the ajax button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- CSS Style** Name of the CSS class, style created in the layout editor.
- Type** Description of the created button.

Code Block

▶ PHP Code	
1	<input type="text"/>

Ajax button coding block.

In this block, you can use macros, Ajax code and JavaScript.

Settings

With this interface, you can set the common attributes of the app.

ATTRIBUTE	VALUE	DESCRIPTION
Application Code	calendar_calendar (9.00.0036)	Application code for ScriptCase internal use.
Description	<input type="text"/>	Application description.
Documents Path	<input type="text" value="/opt/NetMake/v9/wwwroot/scrrip"/>	Absolute path where the application documents are stored.
Image Directory	<input type="text" value="/scriptcase/file/lala"/>	Base directory for the application images.
Application images	<input type="text"/>	Upload here the images that will be used dynamically within the application. Those images will be stored within the images folder and can be used within the events, eliminating the use of the macro sc_image.
Language	<input type="text" value="English (United States)"/>	Language of the generated application.
Share Location Variable	<input checked="" type="checkbox"/>	If you choose the option YES this application will inherit the locale (Language and Regional Settings) from a global variable, otherwise this application will always load its default locale.
Charset	<input type="text"/>	Application specific charset.
Share Theme Variable	<input checked="" type="checkbox"/>	If you choose the option YES this application will inherit the theme from a global variable, otherwise this application will always load its default schema.
Folder	<input type="text" value="root"/>	Folder that will store the application on the working project.
Edit by Project	<input checked="" type="checkbox"/>	Allow other users of the same project to edit the application.
Timeout	<input type="text" value="0"/>	Script execution timeout in seconds. Zero uses the PHP default timeout.
Timeout Message	<input type="text" value="2"/>	Timeout message ajax update.
HelpCase Link	<input type="text"/>	Associate a HelpCase manual to your application.
Maximum file size	<input type="text"/>	Set the upload maximum file size. Same from upload policy: example: upload_max_filesize=512M
Message title	<input type="text" value="{lang_usr_lang_othr_msgs_titl}"/>	Message box title.

Application Settings Interface

- **Attributes**
 - **Application Code** : It is the name that defines an application. An app can be renamed at the [List of Application](#).
 - **Description** : This field contains a brief description of the application objectives.
 - **Documents Path** : The absolute path to store uploaded documents in the application.
 - **Image Directory** : The filesystem directory to store the application images.
 - **Application images** : Import images into the application to allow using them in the application.
 - **Language** : Set the default language of the application. Display all the application hints and messages in the selected language.
 - **Share Location Variable** : Define if the app shares the regional settings with other applications through a session variable.
 - **Charset** : Define a specific charset to use in the application.
 - **Share Theme Variable** : Define if the app shares the Theme settings with other applications through a session variable.
 - **Folder** : Define the project folder that contains the app.
 - **Edit by Project** : Define if other project developers can edit the application.
 - **Timeout** : Set the session runtime timeout in seconds. If the value is Zero, it assumes the default timeout of the PHP.
 - **HelpCase Link** : It allows to associate a [HelpCase](#) file with the application.
 - **Maximum file size** : Set the max size of uploaded files in the application.
 - **Message title** : Message box title.

Error Settings

It groups the notification options of the app.

ATTRIBUTE	VALUE	DESCRIPTION
Error position	Center	Error position in the application
Error Position on the field	Right	Error position when criticizing a field.
Show the Error Title in the Application	<input checked="" type="checkbox"/>	Show the title line of the error message in the application.
Show the Error Title in the Field	<input type="checkbox"/>	Show the title line of the error message in the field.
Error Title	{lang_errm_errt}	Title message of the error
Script Error	<input type="checkbox"/>	Display information about the script and line where the error occurred.
SQL Error	<input checked="" type="checkbox"/>	Display the SQL Select Statement that originated the error.
Debug Mode	<input type="checkbox"/>	Run Application on debug mode, showing SQL commands.
Ajax Error Output	<input checked="" type="checkbox"/>	Window with ajax output, for debug

Error Settings Interface.

- Attributes**
 - Use SweetAlert:** Use the SweetAlert to display messages from the application. When this option is active, it will replace the browser's "confirm" and "alert".
 - SweetAlert position using Toast :** The position to display error messages on the application.
 - Error Position on the field :** The position to display error messages when criticizing the field.
 - how the Error Title in the Application :** Define to display the title line of the error message or not.
 - how the Error Title in the Field :** Define to display the title line of the error message in the field or not.
 - Script Error :** Allows displaying the line code where there is an error..
 - SQL Error :** Allows displaying the SQL statement if it got an error.
 - Debug Mode :** Runs the application in Debug mode, showing all SQL statements the application is executing.
 - Ajax Error Output :** Enables the Ajax alert for debugging errors.

Navigation

This interface allows defining the navigating behavior of the application

ATTRIBUTE	VALUE	DESCRIPTION
Exit URL	<input type="text"/>	URL to redirect the user when exiting the application.
Close on Exit	<input type="checkbox"/>	Close the browser window after exiting the application.
Redirect URL	<input type="text"/>	URL to redirect the user if an application variable is missing.
Redirect Variable	<input type="text"/>	Name of the variable that will contain the URL of the current application upon redirection.
Return After Inserting	<input type="checkbox"/>	Don't open a new blank record after an insert, instead go to edit mode.
Exit Application	<input type="checkbox"/>	Exit the application after processing the form.

Navigation Interface.

Exit URL

URL to where the user goes when he clicks on the "exit" button.

Close on Exit

Close the browser window when the user clicks on the "exit" button.

Redirect URL

Redirect to another URL in case there aren't any global variables available.

Redirect Variable

Creates a variable with the application name and sends it to the redirected application.

Return After Inserting

It opens the inserted record in edit mode instead of to open a new blank form.

Exit Application

It redirects to another application after inserting, updating, or deleting a record.

Messages

On this screen, you can redefine the default application messages to the end-user by customizing the validation messages for Insert, Update, and Delete. You can also define messages for SQL errors and the confirmation outputs.

Messages

Messages		
ATTRIBUTE	VALUE	DESCRIPTION
No Records Message	<input type="text"/>	When the application has no records, it will display this customized text.
Primary key violation	<input type="text"/>	Message to display when the primary key constraint is violated
Unique key violation	<input type="text"/>	Message to display when the unique constraint is violated

No Records Message

The message displayed when the application has no records.

Primary key violation

The message displayed when there is a violation in the database's primary key constraint.

Unique key violation

The message displayed when there is a violation in the database's unique constraint.

Insertion Messages

Insertion messages		
ATTRIBUTE	VALUE	DESCRIPTION
Message After Insert	<input type="text"/>	Message to display after inserting a record.
Message to Confirm Insert	<input type="text"/>	Message to display to confirm inserting of a record.

Message After Insert

The message displayed when inserting a new record.

Message to Confirm Insert

The message displayed if the end-user wants to confirm the inserting of a new record.

Update Messages

Update messages		
ATTRIBUTE	VALUE	DESCRIPTION
Message After Update	<input type="text"/>	Message to display after updating a record.
Message to Confirm Update	<input type="text"/>	Message to display on the update confirmation

Message After Update

The message displayed when updating a record.

Message to Confirm Update

The message displayed if the end-user wants to confirm the changes of a record.

Messages Of Delete

Messages of Delete.		
ATTRIBUTE	VALUE	DESCRIPTION
Message After Delete	<input type="text"/>	Message to display after delete a record.
Message to confirm Delete	<input type="text"/>	Message to display on the delete confirmation

Message After Delete

The message displayed when deleting a record.

Message to confirm Delete

Displays a customized message asking to confirm the record deletion.

If you make no changes, the application will use the default values for the messages. Those values can be configured in [Locales -> Application Language](#).

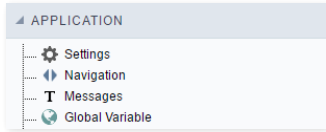
Global Variables

This screen shows all global variables used in the application

The global variable is an external parameter required for running the application. The application can include global variables in the WHERE clause, field definitions and names, event programming, and so on.

You can define global variables by using square brackets ([variable]). You can pass the parameters to the application through one of the methods: Session, Post, and Get

if you need to use [Database Identifiers](#) in or tables, we recommend using the double quotation marks(" ") instead of the brackets([]) because of the conflict with the [Global Variables syntax](#) of Scriptcase. Using the brackets as database identifiers may lead to problems in the Scriptcase applications.



Global variables Interface.

Example:

Select CustomerID, CustomerName from Customers WHERE CustomerID = '[v_customerid]'

We do not recommend using session variables (global variables) with the same name as the table fields.

In this case, the variable **v_customerid** is displaying in the global variables configuration.

ATTRIBUTE	VALUE
global	<div style="border: 1px solid #ccc; padding: 5px;"> <p>Scope</p> <p><input type="checkbox"/> SESSION</p> <p><input checked="" type="checkbox"/> POST</p> <p><input checked="" type="checkbox"/> GET</p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 5px;"> <p>Settings</p> <p><input type="checkbox"/> Optional</p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 5px;"> <p>Type</p> <p><input type="radio"/> Out</p> <p><input checked="" type="radio"/> In</p> </div>

Global variables configuration Interface.

Attribute

It shows the variable's name in the application.

Value

Defines the behavior of the variables, divided into three blocks, they are:

Scope

Defines how the application receives the variable. A variable defined as the POST method in the app does not receive value if it comes with the GET method. Those are the methods available:

- **ession**: Defines that the variable must be created as a PHP session variable by another application.
- **ET**: Defines that the variable must be passed by the URL, that is, visible by the browser.
- **OST**: Defines that the variable must be passed through the POST method of an HTML form.

Settings

- **ptional**: Check it to don't validate the variable at runtime.

Type (In / Out)

Defines whether the variable comes from another application (In), or it is starting in the current app (Out).

Description

Indicates where the application is using the variable.

Synchronize table

This process will performs a comparison between the definitions of application fields and the connected database fields. If there is any change within the connected table, such as adding a new field, exclusion, or a change in the data type, the changes will appear visible like in the image below:

Application Fields	Table Fields
customerid	customerid
companyname	companyname
contactname	contactname
contacttitle	contacttitle
birthdate	birthdate
country	country
regionid	regionid
stateid	stateid
city	city
address	address
postalcode	postalcode
phone	phone
fax	fax
cityid	cityid
creditlimit	creditlimit
cardtype	cardtype
cardnumber	cardnumber
notes	Files

■ Fields that will be created
■ Fields that will be updated
■ Fields that will be removed

Table synchronization interface.

After accessing the function from the link "Synchronize table" you will see a comparison table between the application fields (left table) and the database table fields (right table), like the image above, the fields highlighted in "red" will be deleted of the application, the highlighted fields in "Green" will be inserted in the form, and the highlighted fields in "orange" will be updated, so data type will be updated.

To rename a field in the database table, the table synchronization effect, by comparison, will be the same as deleting a field that existed in the application and the inclusion of a new field.



1. [Home](#)
2. Applications

General Overview

Allows to create link between applications that are in the same project, the link option incorporates various functionalities like above.

New Link

To create a new link you just need to access the option **Link between Applications** located in the menu at the left side.

When clicking in **New Link** the screen below will be display:

When clicking in **New Link** the screen to choose the type of link that will be created will be displayed.

[Edit Link:](#)

Allows to edit registers of the grid using a form.

[Capture Link:](#)

Allows to create a link specifically to the filter's fields, capturing data from other grid.

[Field Link:](#)

Allows to create a link represented through a link, from a grid's column to any application from the project.

[Button Link:](#)

Allows to create a link through grids buttons to any application from the project.



1. [Home](#)
2. [Applications](#)
3. Calendar
4. New link

New link

When you click in “New Link” the ScriptCase application menu will display the screen below.

![[Links menu interface.]]*Links menu interface.*



1. [Home](#)
2. [Applications](#)
3. Calendar
4. Capture link

Capture link



1. [Home](#)
2. [Applications](#)
3. Calendar
4. Link edit

Link edit

In the Link Folder of the Application Menu (Image Below) are displayed the links existing in the application and also the item New Link. When clicking on the existing link it is displayed the screen below that allows to manage the links.

 *Editing Links.*

Actions

Properties Change the link's behavior, position, and how the link opens.

Link Change the application that's being called in the link their parameters.

Delete Remove the existing link.



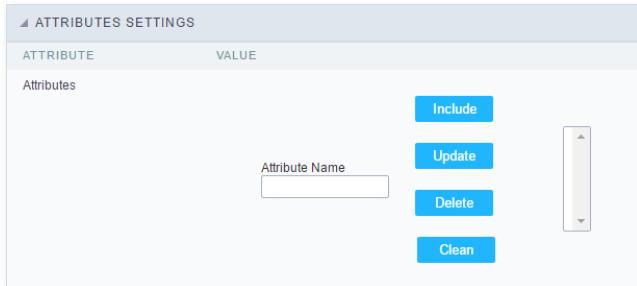
1. [Home](#)
2. [Applications](#)
3. Calendar
4. Button link

Button link

The concept of programming is incorporated in this ScriptCase version. With the use of resources of attributes, methods and libraries. In the previous version was already possible create business rules in the applications, using this concept. The biggest difference is that now this can be realized in a more organized way, making it easier to be developed and comprehended by another developer.

Attributes

The attributes are variables that has the global scope declared in the application. An attribute that can be referenced in all the methods and the application's events.

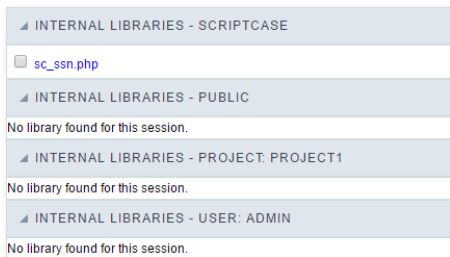


Attributes configuration Interface.

- **attributes**
 - **attribute Name** : This field allows to inform the attributes name, after typing the name just click on **include**. In case you want to **update** click on the attributes name (located on the buttons' right side), do the necessary modifications and click on update. In case it is necessary to **delete** an attribute, it is necessary to do same process as the update, but when selecting the attribute, it is necessary to click on the delete button. The **clear** button, as the name suggests, it clears the list of all the attributes listed on the right.

Internal Libraries

To use the internal libraries just select the correspondent (Project Libraries for example). This process allows the methods existing in the library, to be visible in all the application's Events and Methods. See how to manage the libraries by [clicking here](#).

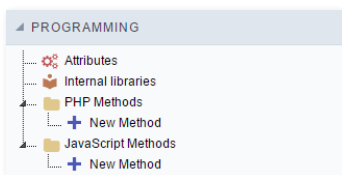


Internal Libraries management Interface

PHP Methods

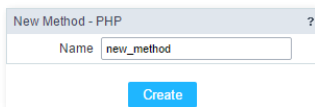
Methods are function or procedures declared by the developer, that helps when applying the routines. Using methods in the applications allows to reuse your code throughout the application, optimizing the development experience.

Creating A New Method



- Give the method a name and click Create. As in the image below.

PHP methods are incorporated into the same class, so it is not possible to use two functions with the same name even when used in different methods.



- Methods can receive parameters.


```
function new_method
1 echo "Hello World!!!";
```

- dd the amount of variables:

Definition of the parameters of the method:

new_method

- No defined parameter.

Add 1 Parameter(s) Cancel

Save

- refining the variables:

Insertion of Parameters

Name	Type	Value Standard
	For Value	
	For Value	
	For References	

Save Back Cancel

- **ame** : Type in the variable's name.
- **ype** : Selecting the type of variables: For Value or For Reference.
- **alue Standard** : The parameter's value used to initialize when calling the method.

References allows to create a second name for a variable that you can use to read and modify the variable's original information.

- diting a parameter:

Definition of the parameters of the method:

new_method

Parameters
Stest = test

☑ ☐ ✎ ✕

Add 1 Parameter(s) Cancel

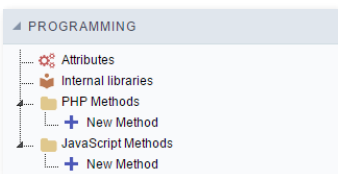
Save

- ☑ : Checking all the parameters in the parameter list.
- ☐ : Uncheck all the parameters in the parameter list.
- ✎ : Edit the selected parameter of the list.
- ✕ : Deletes the selected variable of the list.

JavaScript Methods

Methods are function or procedures declared by the developer, that helps when applying the routines. Using methods in the applications allows to reuse of the your code throughout the application, optimizing the development experience.

Creating A New Method



JavaScript method creation Interface

- refine a name for the method and click on Create. Like the image below.

New Method - JAVASCRIPT ?

Name

Include Method.

- methods can receive parameters.

function new_method

```
1 echo "Hello World!!!";
```

Definition of the parameters of the method:

new_method

No defined parameter.

Parameter(s)

- add the amount of variables:

JavaScript method parameter definition

- defining the variables:

Insertion of Parameters

Name	Type	Value Standard
<input type="text"/>	For Value	<input type="text"/>

Defining the variables

- **name** : Type in the variable's name.
- **type** : Selecting the type of variables: For Value or For Reference.
- **value Standard** : The parameter's value used to initialize when calling the method.

References allows to create a second name for a variable that you can use to read and modify the original variable's information.

- editing a parameter:

Definition of the parameters of the method:

new_method

Parameters

Stest = test

Parameter(s)

- : Checking all the parameters in the parameter list.
- : Uncheck all the parameters in the parameter list.
- : Edit the selected parameter of the list.
- : Deletes the selected variable of the list.

Update applications already created to support Google Calendar

To update the structure and configuration of the application access our article: [Adding new columns in the table for Calendar](#)

Fixing SSL certificate error To correct the error **cURL error 60: SSL certificate problem: unable to get local issuer certificate** access our article: [Calendar displays in execution: cURL error 60](#)

Settings

All the options that can be configured will be reflected in the Calendar Application.

Settings	
ATTRIBUTE	VALUE
Margins	<input type="text"/> Up <input type="text"/> Down <input type="text"/> Right <input type="text"/> Left
Form iframe width	<input type="text" value="0"/>
Form iframe height	<input type="text" value="0"/>
Time format	<input type="text" value="13:00"/>
Start Mode	<input type="text" value="Month"/>
Disable Drag & Drop	<input type="checkbox"/>
Disable option 'Full day'	<input type="checkbox"/>
Minimum time limit	<input type="text"/>
Maximum time limit	<input type="text"/>
Mini calendar and category position	<input type="text" value="Left"/>
Display mini calendar	<input type="checkbox"/>
Full day - hide time	<input type="checkbox"/>
Recurrence - hide fields	<input type="checkbox"/>
Time interval	<input type="text" value="30"/>

Calendar setup screen

- **argins** : Allows you to arrange the display of the application in the page, according to the values entered in the margins. These values must be informed in pixels.
- **orm IFRAME Width** : Allows you to set the form width that will open in the calendar application. These values must be informed in pixels.
- **orm IFRAME Height** : Allows you to set the form height that will open in the calendar application. These values must be informed in pixels.
- **ime Format** : Allows you to choose which time format you want. You can choose between "13:00" (24 hours) or "1:00 pm" (12 AM / PM).
- **art Mode** : Allows you to choose how the calendar will start. The list of events can start in month, day, week, or calendar.
- **isable Drag & Drop** : Allows you to disable the option to drag events to other days or months.
- **isable option 'all day'** : Allows you to disable the 'All Day' option when registering an event. This option is available in the calendar form.
- **inimum Time Limit** : Allows you to limit the minimum time displayed in the Week and Day tabs.
- **aximum Time Limit** : Allows you to limit the maximum time displayed in the Week and Day tabs.
- **ini calendar and category position** : Allows you to position the mini calendar and category in the generated application. This positioning can only be done together to the left or right side.
- **isplay mini calendar** : Allows the display of a mini calendar to navigate independently from the main calendar.
- **ull day - hide time** : Allows you to hide the time fields when the "whole day" is selected in the generated application.
- **ecurrence - hide fields** : Allows you to hide period fields and recurrence information in the generated application.
- **ime interval** : Allows you to enter the time interval you want to display on the calendar in week or day mode.

Calendar Fields

This interface displays the fields relationship that will be used in the application.

You need to relate the table fields to the form application fields.

Calendar Fields Configuration Screen

Google API

The Google Calendar API will allow you to synchronize your Google calendar with the application developed in Scriptcase.

First you must activate the Google Calendar API by clicking on <https://console.developers.google.com/apis/api/calendar-json.googleapis.com/overview>.

After activation you need to configure the "OAuth consent screen" by clicking on <https://console.developers.google.com/apis/credentials/consent>.

OAuth consent screen

- **-mail Address** : Select the email that will be used as the source for the google calendar. The application will use this data for synchronization.
- **product name shown to users** : Allows you to enter the consent name that will be displayed to the clients. Make sure the name accurately describes your product and be careful not to use names that suggest that the product is from Google or another company.
- **ome URL** : Allows you to enter a URL to homepage. (optional)
- **product Logo URL** : Lets you enter a URL that contains an image that will serve as the product logo. (optional)
- **privacy Policy URL** : Lets you enter the URL of the product privacy policy. (optional)
- **terms of Service URL** : Lets you inform the URL of the terms of service of the product. (optional)

Note: You only need to inform the e-mail address and the product name. All other items are optional.

Now you need to create [aCredential](#)> OAuth client ID

Select the "Web Application" option and follow the setup below.

OAuth Client ID Configuration Screen

- **ame**: Enter a name for the OAuth client ID.
- **uthorized JavaScripts Origins** : Insert the root URL of your system. For example: http://yourdomain.com or http://localhost
- **uthorized redirect URLs** : Enter the full URL until your calendar application.

After entering the above data, click "Save."

Now let's configure the Scriptcase application.

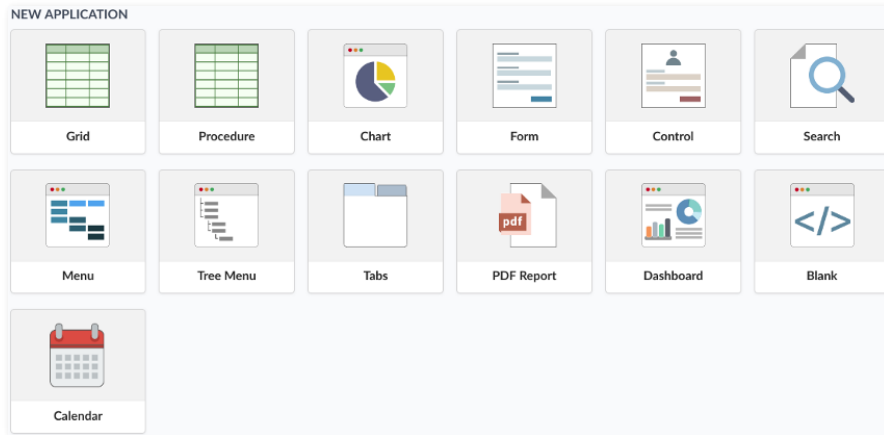
Google Calendar API Settings Screen

- **ntegrate with Google Calendar**: Synchronize with Google calendar events, allowing you to import or export events.
- **Auth Client Json** : Json from Google OAuth with the credential to access the project. To get this content, open the **OAuth Client ID** of the previously configured OAuth_ client and download the JSON.
- **ogin on Google OAuth** : Choose if you want to store the login made in the application to not ask again or if you will request session authentication.
- **ntegrate form** : Enables the form to insert, delete or update events in Google Calendar.
- **nsert** : When inserting an event in the form, it also inserts in google calendar.
- **pdate**: When updating an event on the form, it also updates in google calendar. * **Delete**: When deleting an event in the form, it also excludes in the google calendar.

New Application

The creation process of one application is simple enough and intuitive.

You just need to select which application you want to generate in the applications available in Scriptcase, That it will be generated automatically.



Application Data

When the application is selected, the following screen will be displayed below.

In this table you need to select the connection that will be used in the application. The tables will be loaded and displayed in the field **Table** when the connection is selected.

It is possible to select more than one table in the application creation.

- **connection** - It defines the used connection to create the application. All the project's application will be displayed.
- **ables** - It defines the used tables in the application. (Form and Calendar can only use one table).
- **ields** - It defines the fields that will be part of the applications.
- **QL Select Statement** - Display the select statement created after selecting the tables and fields. This field allows also to insert a previously created SQL, if the used tables are available in the database selected in the connection.

The calendar application is a little different of the others because it has two group of fields, the form's group inside the application and the field's group of the own calendar.

Edit Fields

Fields	Label	Datatype	New	Update	Read-only	Required
id	[lang_calendar_fid_id]	Integer	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
title	[lang_calendar_fid_title]	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
description	[lang_calendar_fid_description]	Text	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
start_date	[lang_calendar_fid_start_date]	Date	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

- **ields** - Name of fields stored in the database.
- **abel** - Name of fields displayed in the generated application.
- **atatype** - Field's datatype.
- **nsert** - It defines the available fields in the insert.

- **pdate** - It defines the available fields in the update.
- **ead-Only** - It defines if the field will be read-only.
- **equired** - It defines if the field will be required to fill.

Edit Calendar Fields

In these field settings, we will relate the calendar fields to the table's fields that we are using in this application.

APPLICATION DATA EDIT FIELDS **EDIT CALENDAR FIELDS** THEME

Make the correlation of the table fields with the fields that will be displayed on the calendar.

FORM FIELDS	TABLE FIELDS
Id *	id ▼
Title *	title ▼
Start date *	start_date ▼
Start time	start_time ▼
End date	end_date ▼

- **orm Fields** - These are the fields returned from the database's table.
- **able Fields** - The calendar fields that are going to be related to the form fields.

Theme

In this screen you select the theme that the application will use. The default project's theme, that can be checked and changed in **Project > Properties**, will be selected automatically.

APPLICATION DATA RELATIONSHIP EDIT FIELDS **THEME**

Sc9_Rhino ▼

Header

|◀ ◀ ▶ ▶| Add Save

Block 1.1

Title 1 Object text

Block 2.1

Title 1	Title 2	Title 3
11111111	22222222	33333333
11111111	22222222	33333333

Footer

- **heme** - List of available themes in the project.

After all the settings are done, click in the button **Create**. In the next screen the settings of the created application will be displayed.

Initial Module

Allows you to set in which mode the application will start when executed.

INITIAL MODULE

Select the Grid module that will be displayed when running the application:

- Search
- Grid
- Summary
- Chart
- Print
- PDF
- Word
- Excel
- RTF
- XML
- CSV


Grid Application Modules


- **ilter** : Application will be started by the filter view, so you can filter the records before the next application, that can be a Grid, PDF, Summary... you can configure that options within the Filter's settings.
- **rid** : Application will be started by the Grid itself, this is the default option.
- **ummary** : Application will be started by the Summary. To use the Summary as initial application it's mandatory to create at least one Group, using the Static Group By settings.
- **hart** : Application will be started by the Chart. To use the Chart as initial application it's mandatory to create at least one Group, using the Static Group By settings.
- **rint** : Application will be started by the print mode, according to the select command.
- **DF** : Application will be started with the option to view or download a PDF file.
- **ord** : Application will be started with the option to view or download a Word file(.doc or .docx).
- **xcel** : Application will be started with the option to view or download an Excel file (.xls or .xls).
- **TF** : Application will be started with the option to view or download a RTF file (.rtf).
- **ML** : Application will be started with the option to view or download a XML file.
- **SV** : Application will be started with the option to view or download a CSV file.


Filter


APPLICATION MODULES


Select the modules that will be part of the Grid and set its parameters:


 Search


 Grid


 Detail


 Summary


 Chart

Filter module	Grid ▼	Application module where search is applied.
Use Iframe	No ▼	Use Iframes to display the Search and the Grid on the same page.
Show Results	Yes ▼	Display the search results on the same page when loading the application for the first time using Iframes.
Iframe Height	1500	Iframe height in pixels where the grid will be displayed.
Search Modal	No ▼	Display the search in a modal window.
Search Modal Height	0	Height search modal.
Search Modal Width	0	Search modal width.
Table Width	0	Width value for the application table.
Table Width Unit	Automatic ▼	Measure unit for the width.

Advanced Settings

Grid Filter Settings


- **ilter Module** : This option configures which application will be called after the Filter.
- **se Iframe** : This option configures the filter to be displayed within an Iframe (displaying the Filter itself and the search results on the same page). This option is available only when the initial module is set to be the Filter.
- **how Results** : Display the search results on the same page when loading the application for the first time when the "Use Iframe" option is set as "Yes", otherwise, the results will be displayed only after the search.
- **frame Height** : Iframe height, in pixels, used to display the search results.
- **earch Modal** : This option configures the Filter Application to open in a modal window.


- **earch Modal Height** : Search Modal box height (in pixels). Option available only when the Search Modal box is enabled.
- **earch Modal Width** : Search Modal box Width (in pixels). Option available only when the Search Modal box is enabled
- **able Width** : Filter application table width. This value can be in percent, pixel or automatic (set by Scriptcase). If you change the value you will also need to set the "Table Width Unit"
- **able Width Unit** : This option configures the unit for the table width (percent, pixel or automatic). The value itself must to be set within the option "Table Width".


Grid


APPLICATION MODULES


Select the modules that will be part of the Grid and set its parameters:


 Search


 Grid


 Detail


 Summary


 Chart

Orientation	<input type="text" value="Horizontal"/>	Grid records' orientation (Horizontal, Vertical, Slide or User Defined).
Fixed label	<input type="text" value="Yes"/>	This option will freeze the column labels on top of the screen during the page scrolling.
Pagination	<input type="text" value="Partial"/>	Pagination method for the grid records.
Lines Per Page	<input type="text" value="10"/>	Number of record lines per page.
Infinite Scroll Increment	<input type="text" value="5"/>	Number of rows to load when loading new records
Open windows using modal	<input type="text" value="Yes"/>	For Grids apps with the Infinite Scroll pagination enabled, this option will set the links between applications and details of the Grid, that are configured to open in iframe, to open using a modal instead.
Maintain records	<input type="text" value="Yes"/>	When you navigate to another window and go back, it will keep the amount records already displayed
Grid height	<input type="text"/>	It defines the height of the Grid in pixels, for when "Infinite Scroll" option is activated. The infinite scrolling will occur for the whole page if this height is not configured.
Table Width	<input type="text" value="100"/>	Set the application width.
Table Width Unit	<input type="text" value="Percent"/>	Measure unit used for the application width.

Advanced Settings


Grid Settings


- **rientation** : This option configures the Grid records' orientation (Horizontal, Vertical, Slide or User Defined). When using the "User Defined" option you can design the HTML manually inside the option "Layout » HTML templates" and select within the Layout settings.
- **ixed Label** : This option will fix the columns' label at the top of the page (it is only available when the Grud records' orientation is set as Horizontal).
- **agination** : This option sets the Grid paging type: Partial (pagination according to the amount of records set per page), Total (displays all records) or Infinite Scroll (automatic strolling according to the increment)
- **ines Per Page** : This option sets the amount of records per page for the Grid
- **nfinite Scroll Increment** : This option sets the number of rows displayed on each increment of new records. It is available only when paging is configured with Infinite Scroll.
- **pen windows using modal** : For Grids Applications with the Infinite Scroll paging enabled, this option will set the links between applications and details of the Grid, that are configured to open in iframe, to open using a modal instead.
- **aintain records** : It configures if the amount of records displayed will be preserved when the application navigates to another window and go back.
- **rid height** : It sets the height of the query in pixels. If it is empty
- **able Width** : Grid application table width. This value can be in percent, pixel or automatic (set by Scriptcase). If you change the value you will also need to set the "Table Width Unit"
- **able Width Unit** : This option configures the unit for the Grid table width (percent, pixel or automatic). The value itself must to be set within the option "Table Width".


Detail


APPLICATION MODULES


Select the modules that will be part of the Grid and set its parameters:


 Search


 Grid


 Detail


 Summary


 Chart

Display Detail	<input type="text" value="In another page"/>	Detail record display mode
Alignment	<input type="text" value="Left"/>	Field alignment for detail page.
Detail Width	<input type="text" value="0"/>	Width value for the application detail
Width of the modal	<input type="text"/>	Width of the modal window in pixels.
Height of the modal	<input type="text"/>	Height of the modal window in pixel
Detail Width Unit	<input type="text" value="Automatic"/>	Measure unit used in the width.

Advanced Settings

Grid Detail settings


- **isplay Detail** : Allows you to set where the detail will open within the Grid Application * **Beside the Grid** : Displays the Grid Details to the right of the record, in the same window where Grid is being displayed. * **Below the Grid** : Displays the Grid Details below the Grid records, in the same window where Grid is being displayed. * **In another page** : Displays the Grid Details in another page, replacing the Grid view. * **In another window** : Displays the Grid Details in a separated browser window. * **Modal** : Opens a pop-up window to the display of the Grid Detail.


- o **ignment** : Using this option you can set an alignment (center, right or left) for the Grid Detail when it's using the "Display Detail" setting as "Beside the Grid" or "Below the Grid".
- o **etail Width** : This option sets the Grid Detail width when it's using the "Display Detail" setting as "Beside the Grid", "Below the Grid", "In another page" or "In another window".
- o **idth of the modal** : Allows you to customize the Grid Detail width when it's using the "Display Detail" option as "Modal".
- o **eight of the modal** : Allows you to customize the Grid Detail height when it's using the "Display Detail" option as "Modal".
- o **etail Width Unit** : This option configures the unit for the "Detail Width" (percent, pixel or automatic).


Summary


APPLICATION MODULES


Select the modules that will be part of the Grid and set its parameters:


 Search


 Grid


 Detail


 Summary


 Chart

This module will not be generated because the grid has no Group By configured.
Create at least one Group By if you want to use this module.

Summary Display
On another page ▾
Summary display page.

[Advanced Settings](#)


Grid Summary Settings


- **ummary Display** : Sets the Grid Summary display option (On another page, On the last page, On every page). The Summary Application Module is only available when a group is created in the Group By settings.


Chart


APPLICATION MODULES


Select the modules that will be part of the Grid and set its parameters:


 Search


 Grid


 Detail


 Summary


 Chart

This module will not be generated because the grid has no Group By configured.
Create at least one Group By if you want to use this module.

Charts display mode	New window ▾	Charts display position.
Display before summary	No ▾	Display chart before summary.
Number of columns	<input type="text" value=""/>	Number of charts per line (one chart in each column).
Margin	<input type="text" value="20"/>	Margin between the charts.
Horizontal alignment	Left ▾	Charts horizontal alignment.
Vertical alignment	Top ▾	Charts vertical alignment.
View settings in the chart window	Yes ▾	Allows you to view the settings in same window as the ch:

[Advanced Settings](#)

Grid Chart settings

- **harts display mode** : Sets the Chart Summary display option (New window, Same page, Other page).The Chart Application Module is only available when a group is created in the Group By settings.
- **isplay before summary** : Sets whether the Chart is displayed above or below the Summary. Available when the Charts display mode is configured in the Same page.
- **umber of columns** : Sets the Chart number of columns per line (one in each column). Available when Charts display mode is configured in the Same page.
- **argin** : Sets the Charts horizontal margin. Available when Charts display mode is configured in the Same page.
- **orizontal alignment** : Sets the Charts horizontal position (Left, Right, Center). Available when Charts display mode is configured in the Same page.
- **ertical alignment** : Sets the Charts vertical position (Top, Center, Bottom). Available when Charts display mode is configured in the Same page.
- **iew settings in the chart window** : Sets the chart settings display on the same page. Available when the Char display mode is configured on Other page or new window.

GRID SETTINGS	
ATTRIBUTE	VALUE
Friendly URL	<input type="text"/>
Display Line Number	<input type="checkbox"/>
Display Titles	<input checked="" type="checkbox"/>
Line break in title	<input type="checkbox"/>
Horizontal Alignment	Center ▾
Margins	<input type="text"/> Up <input type="text"/> Down <input type="text"/> Right <input type="text"/> Left
Alignment	Left ▾
Table Columns	Automatic ▾
Refresh Interval	<input type="text" value="0"/>

Grid application advanced settings

- **riendly URL** : This field allows you to change the URL that will be called by the application. Allowed characters are the same available on URLs: a-z, A-Z, 0-9, -_. This option can also be changed on the home screen, on the "Friendly URL" column at the applications list.
- **isplay Line Number** : This option sets whether to display or not the sequence number of each Grid row.
- **isplay Titles** : This option sets if the column title (column label) will be displayed or not.
- **ine break in title** : This option sets if the column line title will break or not.
- **argins** : Sets the application margins in pixels (up, down, right and left).
- **ignment** : Allows you to set the fields alignment for when the Grid orientation is set as Vertical or Slide.
- **able Columns** : Sets the column widths type: Provides (It will assume the informed width values at field level in the configuration of the visualization), Calculated (Calculate size according to type and the field size) and Automatic (The alignment will be according to the browser criteria).
- **efresh Interval** : Allows you to set a reload interval for the page, in seconds. When is set as zero, there will be no page reload.

This interface is useful for editing the field settings and their position to display.

Fields	Label	Datatype	Line Break	Line Break PDF	Title Horizontal Alignment	Text Alignment	Font Color
PAGE: PAG1							
BLOCK: GRID_ORDERS							
 orderid	Orderid	Integer	<input type="checkbox"/>	<input type="checkbox"/>	Right	Right	<input type="text"/>
 customerid	Customer Name	Text	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Left	Left	<input type="text"/>
 employeeeid	Employeeid	Integer	<input type="checkbox"/>	<input type="checkbox"/>	Right	Right	<input type="text"/>
 orderdate	Orderdate	Date	<input type="checkbox"/>	<input type="checkbox"/>	Center	Center	<input type="text"/>
 requireddate	Requireddate	Date	<input type="checkbox"/>	<input type="checkbox"/>	Center	Center	<input type="text"/>
 shippeddate	Shippeddate	Date	<input type="checkbox"/>	<input type="checkbox"/>	Center	Center	<input type="text"/>

Fields

Allows accessing the field settings (pencil icon on the left). You can change the field position by dragging them to the desired position. Drag a field to “fields not displayed” if you don’t want it in the app.

Label

Defines the title of a field in the app. For example: if the field name in the database is fld_txt_customer_name, you can display the label “Customer Name”.

Data Type

It informs the data type of the field.

Line Break

Allows the Line Break in the records when the field text is greater than the column width.

Line Break PDF

Allows the Line Break when the user exports the app as a PDF.

Title Horizontal Alignment

Defines the horizontal alignment of the field label.

Text Alignment

Defines the horizontal alignment of the field text.

Font Color

Configure the text color for the field.

This interface allows to sort the fields positioning.

Select the fields that will be displayed.

orderid customerid employeeid orderdate requireddate shippeddate shipvia freight priceorder shipcountry shipregion shipstate shipcity shipname shipaddress shippostalcode	➤ ➤ ➤ ➤ ➤	Pag1 form_orders orderid customerid employeeid orderdate requireddate	⤴ ⤴ ⤴
--	-----------------------	---	-------------

Save Restore

Field Positioning

The left box has a list of all fields available in the application. Here you must select which fields you want to display in the app.

The right box contains the selected fields displayed in the application.

Use the arrows buttons between the boxes to select or deselect a field.

The application toolbar has two segments: Top and Bottom, in a way that is possible to define to display buttons into both areas. Those areas work independently, allowing them to display the same button, for example.

It's also possible to select the buttons and their position if the application is running on a mobile device.

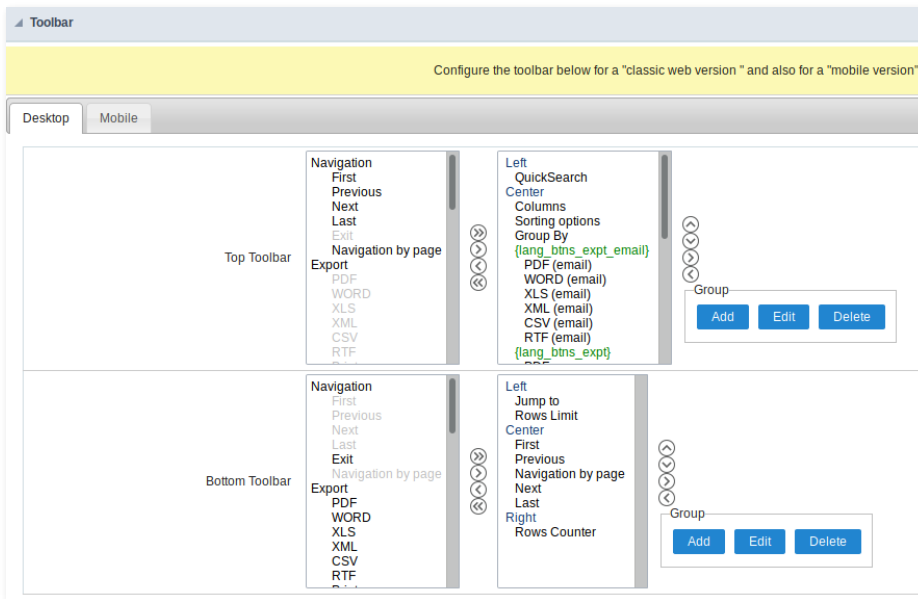
Toolbar

Desktop

Here we must inform the toolbar settings for the "Classic Web Version" mode and which buttons are available in the application when accessed from a **Desktop** environment.

Mobile

Here we must inform the toolbar settings for the "Mobile Version" mode. That is which buttons are available in the application when accessed from a **Mobile** dispositive.



Navigation:

Buttons relative to the navigation of the application.

Next	Move to the next page that can be a single record or a list of records.
Previous	Returns to displays the previous page records or a single record.
First	Move to the First page or record
Last	Move to the Last page or record
Exit	Close the application
Navigation by page	Displays a "page-number" navigation bar. Example: 1 2 3 4 5

Export:

Groups the options relative to the generated exports. Scriptcase generates the following export formats:

PDF	Generates a complete Report with all the data of the application in a PDF format.
WORD	Generates a complete Report with all the data of the application in a WORD format.
XLS	Generates a complete Report with all the data of the application in an EXCEL format.
XML	Generates a complete Report with all the data of the application in an XML format.
CSV	Generates a complete Report with all the data of the application in a CSV format.
RTF	Generates a complete Report with all the data of the application in an RTF format.
Print	Creates an HTML with the records ready for printing.

Export by Email:

Groups the options relative to the emails exports. Scriptcase generates the following export formats:

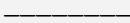
PDF (Email)	Sends by Email a complete report with all the data of the application in a PDF format.
WORD (Email)	Sends by Email a complete report with all the data of the application in a WORD format.
XLS (Email)	Sends by Email a complete report with all the data of the application in an EXCEL format.
XML (Email)	Sends by Email a complete report with all the data of the application in an XML format.
CSV (Email)	Sends by Email a complete report with all the data of the application in a CSV format.
RTF (Email)	Sends by Email a complete report with all the data of the application in an RTF format.

Others:

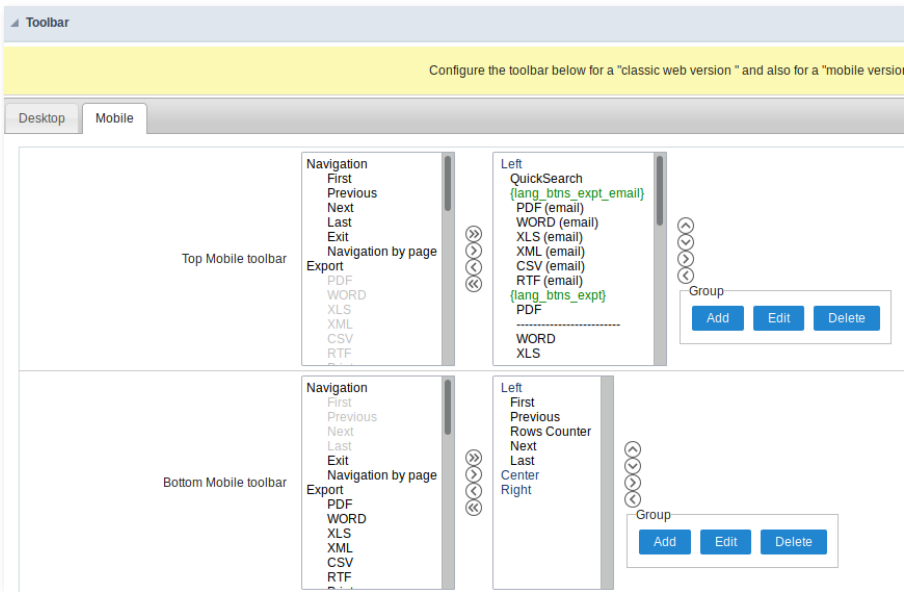
Other options available in the Grid application.

Jump to	Move to the informed page.
Rows Limit	It is a Combobox that defines the number of rows per page.
Search	Goes to the Search Form to filter the records.
Dynamic Search	It displays the fields of the search to filter the records.
Columns	Allows to include or remove columns of the Grid on the fly.
Sorting Options	Allows selecting the order of the records based on the field.
Group By	Allows to select or change a Group By rule on the Grid.
Save Grid	Allows saving the current state of the application. For example, in the advanced search, you can save the search data for further use.
Quick Search	Allows to perform a quick search in the records of the application.
Gantt	It displays a Gantt chart, if it was previously set.
Summary	It displays a summary with the synthetical data of the records.
Form Buttons	It displays the Form Buttons when there's an Application Link from the Grid to a Form.
Languages	It displays a Combobox with the names available, defined in the project properties.
Themes	It displays a Combobox with the themes available, defined in the project properties.
Rows Counter	It displays the number of records retrieved by the application.
HelpCase	Displays a button to open the help page.

Separator:

	Displays a line separating the buttons, when used the Group Buttons.
--	--

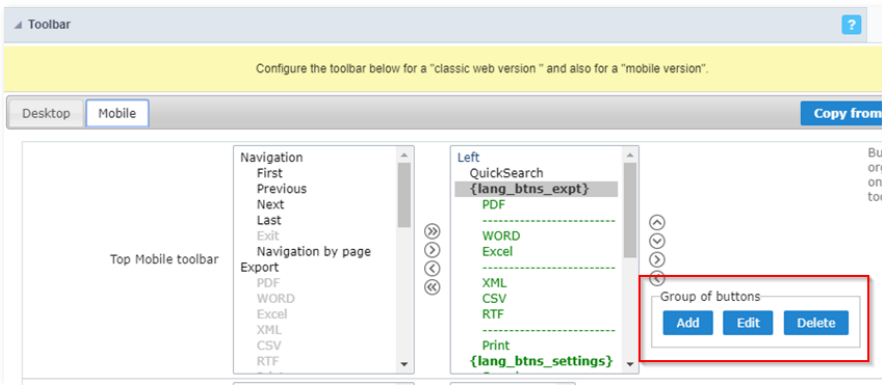
Toolbar Mobile



Contains the same options as the **Desktop** version, adding only the item "Copy from Desktop", on which, when clicked, it copies the items from the **Desktop** toolbar to the **Mobile** toolbar.

Buttons Group

The **Group** option allows you to group a set of buttons of the application toolbar to display them as a dropdown, for example.



Add
Add a new group of buttons.

Edit
Edit an existent group of buttons.

Delete
Delete the selected group of buttons.

When you press the **Add** or **Edit** option, you can see the settings to configure the grouper:

Edit

DISPLAY AS
 Dropdown

DROPDOWN LIST THEME NAME
 Application Theme group_4

LABEL HINT \ TITLE
 Group

IMAGE BUTTON TYPE
 Button

DISPLAY DISPLAY POSITION
 Only Text Text on the right

Cancel Add

Display As

Allows displaying the group button as **Dropdown** or **Side by Side**.

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print Exit

Customer ID	Company Name	Phone	Credit Limit
		30074321	\$3,367.41
		55554729	\$7,371.95
		55553932	\$6,757.53

A GRID WITH VARIOUS EXPORTATION FORMATS

PDF WORD Excel XML CSV RTF Print Exit

Dropdown List Theme

Allows defining the Dropdown theme selecting between **Application theme** and **Button theme**.

Name

Allows defining a name for the button group.

Label

It is the displayed name for the button group in the application.

Hint\Title

Displays a hint to the end-user when the mouse is on the group of buttons.

Button Type

Allows displaying the button group as a Button, Image, or Link.

Image

Allows selecting an image for the button.

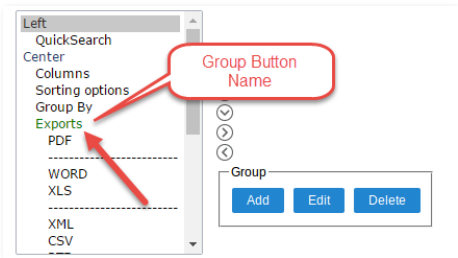
Display

Defines if the button displays only Text, only image, or both.

Display Position

Defines the position of the Text or Image (Text to the right, Image to the right).

After creating a button group, you need to move the grouped buttons below of the Button Group and then move them to the right. Like the image below:



Buttons Settings

Button	Label	Hint
QuickSearch		
Dynamic Search		
Insert		
Cancel		
Update		
Delete		
Exit		
Jump to		
First		
Previous		
Navigation by page		
Next		
Last		
Rows Counter		
New		

Button:
It displays the buttons available in the application.

Label:
Allows defining the label of the buttons to display for the users.

Hint:
Allows defining the buttons hint to display for the users.

Application Hotkeys

Scriptcase allows creating shortcut keys to your applications. You can select a predefined template or create specific actions for an application.

VALUE	DESCRIPTION				
<input checked="" type="checkbox"/> toggleHotkeys();	Define if the application will use hotkeys				
SC_DefaultHotkeys	Select the hotkey template from previously created schemas				
<input type="button" value="Clear"/> <input type="button" value="+"/>					
<table border="1"> <thead> <tr> <th>ACTION</th> <th>KEYBINDING</th> </tr> </thead> <tbody> <tr> <td colspan="2" style="text-align: center;">No hotkeys configured</td> </tr> </tbody> </table>		ACTION	KEYBINDING	No hotkeys configured	
ACTION	KEYBINDING				
No hotkeys configured					
<input type="button" value="Clear"/> <input type="button" value="+"/>					

Use hotkeys
Defines if the application uses hotkeys. When you enable this option, the default shortcut keys settings are disabled.

Hotkeys templat
Select the [hotkey template](#) previously created.

Action
Selects the triggered action when pressing the selected key.

Keybinding
Selects the keys responsible for executing the chosen action.

Add "+"

Adds a new action on the keys list.

Clear

It clears the selected hotkeys preference.

Options

ATTRIBUTE	VALUE
Display Summary	<input type="checkbox"/>
Format Row Counter	<input type="text" value="{lang_othr_smry_info}"/>
The number of links displayed	<input type="text" value="5"/>
Jump to	<input type="text" value="Page"/>
Records per page	<input type="text" value="10,20,50"/>
Toolbar buttons	<input type="text" value="Modal"/>

Display Summary:

Allows to display the summary button or not.

Row Counter:

Allows to display the Row Counter or not.

Example: (1 to 10 of 200)

The Number of Links Displayed:

Defines the number of links per page, when the navigation option is disabled.

Jump To:

Move to the informed record page.

Records by page:

Allows defining the number of records to be exhibited for each page. To show all records use the option "all".

Example: 10,20,30, all

Toolbar Buttons:

The type of view for the Toolbar buttons (A DIV below the toolbar, Modal).

PDF Settings

WORD Settings

General settings	
ATTRIBUTE	VALUE
Word configurable	<input type="checkbox"/>
Grid Export modules	<input checked="" type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Summary export modules	<input type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Export with password	<input type="checkbox"/>
Select columns	<input type="checkbox"/>
Open WORD Directly	<input type="checkbox"/>

Word configurable

It allows the end-user to configure the export parameters on the fly.

Export Modules of Grid

This option allows the developer to choose which modules are available in the grid export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
---------------------	---

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Summary Export Modules

This option allows the developer to choose which modules are available in the summary export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid
	<input type="checkbox"/> Summary
	<input checked="" type="checkbox"/> Chart

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Export with password

This option allows the developer to set a password to export the file.

Export with password	<input checked="" type="checkbox"/>
Export with password	<input type="text"/>

Password

It allows the developer to set the requested password when exporting the file.

Runtime Example:

PDF settings	
GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Print settings	Color
Page Layout	A4 (210 X 297 mm)
Orientation	Portrait
Other options	<input checked="" type="checkbox"/> Generate Bookmarks <input checked="" type="checkbox"/> Displays the header on all pages <input checked="" type="checkbox"/> Displays the title on all pages
Password

Select columns

It allows the user has the option to choose the columns he wants to export at runtime.

PDF settings	
Select Columns	
	<input type="checkbox"/> Accountid <input type="checkbox"/> Accountdescription <input type="checkbox"/> Accountorder

Open WORD Directly

Opens the generated WORD file without the need to display an intermediate page with a link to download it.

WORD Settings

CSV Settings

General settings	
ATTRIBUTE	VALUE
Configurable CSV	<input type="checkbox"/>
Grid Export modules	<input checked="" type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary
Summary export modules	<input type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary
Export with password	<input type="checkbox"/>
Select columns	<input type="checkbox"/>
Open CSV Directly	<input type="checkbox"/>

Configurable CSV

It allows the end-user to configure the CSV export parameters on the fly.

Export Modules of Grid

This option allows the developer to choose which modules are available in the grid export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid
	<input type="checkbox"/> Summary
	<input checked="" type="checkbox"/> Chart

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Summary Export Modules

This option allows the developer to choose which modules are available in the summary export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid
	<input type="checkbox"/> Summary
	<input checked="" type="checkbox"/> Chart

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Export with password

This option allows the developer to set a password to export the file.

Export with password	<input checked="" type="checkbox"/>
Export with password	<input type="text"/>

Password

It allows the developer to set the requested password when exporting the file.

Runtime Example:

PDF settings	
GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Print settings	Color
Page Layout	A4 (210 X 297 mm)
Orientation	Portrait
Other options	<input checked="" type="checkbox"/> Generate Bookmarks <input checked="" type="checkbox"/> Displays the header on all pages <input checked="" type="checkbox"/> Displays the title on all pages
Password

Select columns

It allows the user has the option to choose the columns he wants to export at runtime.

PDF settings	
Select Columns	
	<input type="checkbox"/> Accountid <input type="checkbox"/> Accountdescription <input type="checkbox"/> Accountorder

Open CSV Directly

Opens the generated CSV file without the need to display an intermediate page with a link to download it.

CSV Settings

CSV Settings	
ATTRIBUTE	VALUE
Lines Separator	CRLF
Columns Separator	:
Text Delimiter	"
Add Label	<input type="checkbox"/>

Line separator

It allows you to define the line separator character, each line representing a record.

Column separator

It allows you to define the column separator character, each column representing a database field.

Text delimiter

It allows you to define the character used to delimit the text of the columns when we have database fields of type String.

Add label

It allows you to define displaying the label of the columns in the first line of the file.

XLS Settings

General settings	
ATTRIBUTE	VALUE
Export with password	<input type="checkbox"/>
Select columns	<input checked="" type="checkbox"/>
Open Excel Directly	<input type="checkbox"/>

Export with password

This option allows the developer to set a password to export the file.

Export with password	<input checked="" type="checkbox"/>
Export with password	<input type="text"/>

Password

It allows the developer to set the requested password when exporting the file.

Runtime Example:

PDF settings	
GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Print settings	Color
Page Layout	A4 (210 X 297 mm)
Orientation	Portrait
Other options	<input checked="" type="checkbox"/> Generate Bookmarks <input checked="" type="checkbox"/> Displays the header on all pages <input checked="" type="checkbox"/> Displays the title on all pages
Password

Select columns

It allows the user has the option to choose the columns he wants to export at runtime.

PDF settings	
Select Columns	
	<input type="checkbox"/> Accountid <input type="checkbox"/> Accountdescription <input type="checkbox"/> Accountorder

XLS Configurable

It allows the end-user to configure the XLS export parameters on the fly.

Open XLS Directly

It allows the end-user to configure the XLS export parameters on the fly.

XLS Settings

Default values settings and itens to end-user export interface

ATTRIBUTE	VALUE	DESCRIPTION
Excel Configurable	<input checked="" type="checkbox"/>	Allows the user to configure the parameters of creation of Excel during the execution of the application.
Excel settings available to the end-user.		
Grid Export modules	<input checked="" type="checkbox"/>	Defines which modules will be exported in the Grid.
Summary export modules	<input checked="" type="checkbox"/>	
Format	xlsx	Excel format.
Export with totals	<input checked="" type="checkbox"/>	Enables the display of the totals when exporting to Excel.

Export Modules of Grid

This option allows the developer to choose which modules are available in the grid export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules

- Grid
- Summary
- Chart

You can see the Summary option disabled at running time.

GENERAL

Select the modules Grid Summary Chart

Summary Export Modules

This option allows the developer to choose which modules are available in the summary export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules

- Grid
- Summary
- Chart

You can see the Summary option disabled at running time.

GENERAL

Select the modules Grid Summary Chart

Format

It allows defining the Excel format of the generated document (xls or xlsx).

Export with totals

Show the totals when exporting the application data to Excel.

XML Settings

General settings	
ATTRIBUTE	VALUE
Configurable XML	<input type="checkbox"/>
Grid Export modules	<input checked="" type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary
Summary export modules	<input type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary
Export with password	<input type="checkbox"/>
Select columns	<input type="checkbox"/>
Open XML Directly	<input type="checkbox"/>

Configurable XML

It allows the end-user to configure the XML export parameters on the fly.

Export Modules of Grid

This option allows the developer to choose which modules are available in the grid export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
---------------------	---

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Summary Export Modules

This option allows the developer to choose which modules are available in the summary export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

Grid Export modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
---------------------	---

You can see the Summary option disabled at running time.

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Export with password

This option allows the developer to set a password to export the file.

Export with password	<input checked="" type="checkbox"/>
Export with password	<input type="text"/>

Password

It allows the developer to set the requested password when exporting the file.

Runtime Example:

PDF settings	
GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Print settings	Color
Page Layout	A4 (210 X 297 mm)
Orientation	Portrait
Other options	<input checked="" type="checkbox"/> Generate Bookmarks <input checked="" type="checkbox"/> Displays the header on all pages <input checked="" type="checkbox"/> Displays the title on all pages
Password

Select columns

It allows the user has the option to choose the columns he wants to export at runtime.

The 'Select Columns' dialog box is shown with a dashed border. It contains a list of columns: Accountid, Accountdescription, and Accountorder. The 'Accountid' column is highlighted, indicating it is selected for export.

Open XML Directly

Opens the generated XML file without the need to display an intermediate page with a link to download it.

XML Settings

The XML Settings panel is shown with a table structure:

ATTRIBUTE	VALUE
Use Label on the Tag	<input type="checkbox"/>
XML Format	<input type="radio"/> Attr <input checked="" type="radio"/> Tag

Use Label on the Tag

It allows using the field labels on Tags of the XML file.

XML Format

Define how to generate the records in the XML file, if it will store the values on attributes or new elements. (Attr or Tag)

Print Settings

The General settings panel is shown with a table structure:

ATTRIBUTE	VALUE
Configurable Print HTML	<input type="checkbox"/>
Grid Export modules	<input checked="" type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Summary export modules	<input type="checkbox"/> Grid <input checked="" type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Select columns	<input type="checkbox"/>

Configurable Print HTML

It allows the end-user to configure the HTML Printing parameters on the fly.

Export Modules of Grid

This option allows the developer to choose which modules are available in the grid export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

The Grid Export modules selection panel is shown with a table structure:

Grid Export modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
---------------------	---

You can see the Summary option disabled at running time.

The GENERAL settings panel is shown with a table structure:

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Summary Export Modules

This option allows the developer to choose which modules are available in the summary export. The user can select the modules at runtime.

Example:

Disabling the Summary option in the development environment.

The Grid Export modules selection panel is shown with a table structure:

Grid Export modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
---------------------	---

You can see the Summary option disabled at running time.

The GENERAL settings panel is shown with a table structure:

GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart

Export with password

This option allows the developer to set a password to export the file.

Export with password	<input checked="" type="checkbox"/>
Export with password	<input type="text"/>

Password

It allows the developer to set the requested password when exporting the file.

Runtime Example:

PDF settings	
GENERAL	
Select the modules	<input checked="" type="checkbox"/> Grid <input type="checkbox"/> Summary <input checked="" type="checkbox"/> Chart
Print settings	Color
Page Layout	A4 (210 X 297 mm)
Orientation	Portrait
Other options	<input checked="" type="checkbox"/> Generate Bookmarks <input checked="" type="checkbox"/> Displays the header on all pages <input checked="" type="checkbox"/> Displays the title on all pages
Password

Select columns

It allows the user has the option to choose the columns he wants to export at runtime.

PDF settings	
Select Columns	
<input type="checkbox"/>	Accountid
<input type="checkbox"/>	Accountdescription
<input type="checkbox"/>	Accountorder

RTF Settings

General settings	
ATTRIBUTE	VALUE
Select columns	<input type="checkbox"/>
Open RTF Directly	<input type="checkbox"/>

Select columns

This option allows the end-user to select which fields he wants to export to the file.

PDF settings	
Select Columns	
<input type="checkbox"/>	Accountid
<input type="checkbox"/>	Accountdescription
<input type="checkbox"/>	Accountorder

Open RTF Directly

Opens the generated RTF file without the need to display an intermediate page with a link to download it.

Word Settings

WORD Settings	
ATTRIBUTE	VALUE
Print Type	Color
Rows per Page in the Grid	0
Rows per Page in Summary	0

Word export settings

Print Type

Allows you to set the print mode for the word file (Both, Black & White, Color).

Rows per Page in Grid

Allows you to set the number of lines per page of the query will be displayed in the file.

Rows per Page in Summary

Allows you to set the number of lines per page in the Summary that be displayed in the file.

Open Word Directly

Allows you to set whether the document will be generated directly or if an intermediate page will be displayed.

CSV Settings

ATTRIBUTE	VALUE
Lines Separator	CRLF
Columns Separator	;
Text Delimiter	"
Add Label	<input type="checkbox"/>

CSV export settings

Lines Separator Allows you to set the separator for character.

Columns Separator Allows you to set the column separator for character and records.

Text Delimiter Allows you to define the character used to delimit the text columns.

Open CSV Directly Allows you to set whether the document will be generated directly or if an intermediate page will be displayed.

Add Label Allows you to define if the columns label will be added to the file.

XLS Settings

ATTRIBUTE	VALUE	DESCRIPTION										
Excel Configurable	<input checked="" type="checkbox"/>	Allows the user to configure the parameters of creation of Excel during the execution of the application.										
Excel settings available to the end-user.												
<table border="1" style="width: 100%;"> <thead> <tr> <th>Grid Export modules</th> <th>Options</th> </tr> </thead> <tbody> <tr> <td>Summary export modules</td> <td>Grid Export modules</td> </tr> <tr> <td>Format</td> <td>Summary export modules</td> </tr> <tr> <td>Export with totals</td> <td>Format</td> </tr> <tr> <td></td> <td>Export with totals</td> </tr> </tbody> </table>			Grid Export modules	Options	Summary export modules	Grid Export modules	Format	Summary export modules	Export with totals	Format		Export with totals
Grid Export modules	Options											
Summary export modules	Grid Export modules											
Format	Summary export modules											
Export with totals	Format											
	Export with totals											
Grid Export modules	<input checked="" type="checkbox"/>	Grid Summary										
Summary export modules	<input checked="" type="checkbox"/>	Grid Summary										
Format	xlsx	Excel format.										
Export with totals	<input checked="" type="checkbox"/>	Enables the display of the totals when exporting to Excel.										

XLS export settings

Open XLS Directly Allows you to set whether the document will be generated directly or if an intermediate page will be displayed.

Format Allows you to define the format of the generated document (xls or xlsx).

Print Settings

ATTRIBUTE	VALUE
Print Mode	Both
Print Type	Both
Rows per Page	0
Rows per Page in Summary	0
Print Background	<input checked="" type="checkbox"/>

Print export settings

Print Mode Allows you to define the contents of the file printing (Both, Current Page or Full Report).

Print Type Allows you to set the print mode for the file (Both, Black & White, Color).

Rows per Page Allows you to set the number of lines per page of the query will be displayed in the file.

Rows per Page in Summary Allows you to set the number of lines per page in the Summary that be displayed in the file.

Print Background Allows you to define whether the background will appear for printing.

Other Formats

General settings	
ATTRIBUTE	VALUE
Select columns	<input type="checkbox"/>
Open RTF Directly	<input type="checkbox"/>

XML and RTF export settings

Generate XML directly Allows you to set whether the document will be generated directly or if an intermediate page will be displayed.

Generate RTF directly Allows you to set whether the document will be generated directly or if an intermediate page will be displayed.

SQL Settings

This interface allows configuring the related database settings, such as the SQL statement, the used database connection, case sensitive, and others.

ATTRIBUTE	VALUE	DESCRIPTION
SQL Select Statement	<pre>SELECT customerid, companyname, contactname, contacttitle, birthdate, country, regionid, stateid, city,</pre>	
Limit	<input type="text"/>	It sets the number of records to be retrieved from the SQL statement.
SQL Preparation	<input type="text"/>	
Connection	<input type="text" value="conn_example"/>	Connection name to access the database.
Use Customized Message	<input type="checkbox"/>	Use a customized error message when the application has no records.
No Records Message	<input type="text"/>	When the application has no records, it will display this customized text.
Font	<input type="text" value="Aa"/>	Font face of the error message.
Font Size	<input type="text" value="12"/>	Font size of the error message.
Font Color	<input type="text" value="#000000"/>	Font color of the error message.
Variable for Table	<input type="text"/>	Variable name used for replacing the table name. Please indicate the name of the table that will be replaced by the variable value.
Fields Variables	<div style="border: 1px solid black; padding: 2px;"> <p>Variable</p> <input type="text"/> <p>customerid</p> </div>	Variables for substitution of the field names on the application. For each dynamically determined field, inform the name of the variable and the field that will be substituted.
Case Sensitive	<input checked="" type="checkbox"/>	Use case sensitive.

Grid SQL configuration

SQL Select Statement

It allows you to define the primary SQL of the application. You can edit this SQL to add or delete fields.

Limit

Lets you limit the display in the number of records retrieved by SQL query.

SQL Preparation

You can enter SQL commands or procedure names to execute them before the primary SQL of the application.

Connection

It allows defining the database connection of the application. You can change the connection to another one that has the same table.

Use Customized Message

Lets you define to display the "no records" message or not.

No Records Message

Lets you set the message when the application has no records.

Font

This option is available when using the "Use Customized Message" option. It lets you set the font for the message.

Font Size

This option is available when using the "Use Customized Message" option. It lets you to set the font size.

Font Color

This option is available when using the "Use Customized Message" option. It lets you to set the font color.

Variable for Table

It allows to use a variable to change a part of the string containing the table name.

Fill the first input with the name of the variable (replace to). The second input you should fill with the part of the name of the table you want to replace (replace from).

Fields Variables

It allows to use a variable to change a part of the string containing the field name.

Fill the first input with the name of the variable (replace to). The second input you should fill with the part of the name of the field you want to replace (replace from).

Case sensitive

It defines if the database connection uses case sensitive or not.

Security ?		
ATTRIBUTE	VALUE	DESCRIPTION
Use Security	<input type="checkbox"/>	Application needs validation to run. Read about the sc_apl_status macro.
Url output of the security	<input type="text"/>	Uri to be redirected if the user does not have access to this application.
Use Password	<input type="checkbox"/>	Enter password required to run the application.
Request password just once	<input type="checkbox"/>	Request password just once per session without asking for password every time you open the application in the same session.
Enable direct call by URL	<input checked="" type="checkbox"/>	Allow an application to be called by typing the URL directly in the browsers.

Application Security Configuration

Use Security

Turns the application accessible just when using the macro sc_apl_status or using the security module. When it is enabled, the application displays a message "User not Authorized" if the user tries to access it directly.

Url output of the security

It redirects the application to another URL after displays the warning "User not Authorized".

Use Password

Turns the application accessible just by using a password.

Request password just once

Request the password once per session, not needing to inform the password every time you access the application.

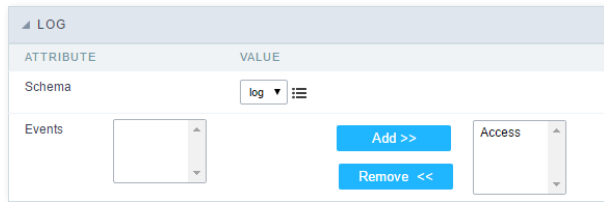
Enable direct call by URL

Allows to call the application directly from the URL in the browsers.

Allows calling the application directly from the URL in the browsers. In the development environment, you can disable those options for testing, it avoids requiring the login every time you run the application. To disable the security and password in development, you must access Options > My ScriptCase and uncheck the options: Enable Use of Security, Enable Use of Password.

This interface allows you to define a Log schema to the app. The Log schema tracks what the end-users are doing in the app. Those are the events it can tracks: Access, Insert, Update, and delete.

To learn how to create a Log schema [see here](#).



The screenshot shows a configuration window titled "LOG". It features a table with two columns: "ATTRIBUTE" and "VALUE". The "Schema" attribute has a dropdown menu currently set to "log". Below the table, there are two empty list boxes: "Events" on the left and "Access" on the right. Between these two boxes are two blue buttons: "Add >>" and "Remove <<".

Application Log configuration

Schema Select an existing log schema for the application.

Events You can select just specific events to save in the log.

1. [Home](#)
2. Applications

General Overview

The applications can have two types of fields.

Table Fields

Columns of the table connected to the application and their data types.

Example:

In a table with the following structure:

```
CREATE TABLE `orders` (
  `orderid` int(11) NOT NULL,
  `customerid` varchar(5) NULL DEFAULT NULL,
  `employeeid` int(11) NULL DEFAULT NULL,
  `orderdate` date NULL DEFAULT NULL,
  `requireddate` date NULL DEFAULT NULL,
  PRIMARY KEY (`orderid`) USING BTREE
);
```

These are the listed fields on left menu **Fields**:

Virtual Fields

Fields created inside application to assist in the development process.

How to create a virtual field

1. Inside an application, access the left menu **Fields** and click on **New Field** option.
2. Choose how many fields do you want to create:

1. Create the fields and define type, name and label for each one:

- **Type:** Data type of the field.
- **Name:** Internal name of the field. Used to identify the field on events and Scriptcase interface.
- **Label:** Title of the field displayed on the application.

Scriptcase **doesn't insert** virtual fields on the table after its creation.

Types of fields by applications

Application Table Fields Virtual Fields

Grid	<input type="text"/>	<input type="text"/>
Procedure	<input type="text"/>	<input type="text"/>
Chart	<input type="text"/>	<input type="text"/>
Form	<input type="text"/>	<input type="text"/>
Control	<input type="text"/>	<input type="text"/>
Search	<input type="text"/>	<input type="text"/>
PDF Report	<input type="text"/>	<input type="text"/>
Calendar	<input type="text"/>	<input type="text"/>

The applications **Menu**, **Tree Menu**, **Dashboard** and **Blank** were not listed because they haven't manage fields.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. Text

Text

General Settings

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced by the curly brackets {}. While running the application, the field within the curly brackets is replaced by its value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Martial Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute Value Lookup Description

1	Sports
2	Culture
4	Pleasure
8	Reading
16	Music

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in :**
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

///
PDF Configuration {#id-04}

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".



Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout



Chart layout settings interface.

1. [Home](#)
2. [Applications](#)
3. Stored Procedures
4. Multiple Lines Text

Multiple Lines Text

General Settings

Multiple Lines Text field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Multiple Lines Text, it accepts letters, numbers and special characters in multiple lines.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Case Settings** : Convert the letter from the field when losing focus. The options are:
 - **Upper case** : All in Upper Case
 - **Lower case** : All in Lower case
 - **Capitalize first word** : Capitalizes the first letter of the first word
 - **Capitalize all words** : Capitalizes the first letter of all the the words
- **Show HTML content** : Determines if the HTML contained in the field will be displayed or not. If enabled, the HTML will be displayed, otherwise the HTML will be interpreted by the browser.
- **Grid Mask** : Defines the mask for the field display. There are two typed of masks described below:

Character	Description
X	Placeholder to any character. Replaced by any character. If number of characters entered are less then the mask size, the field value is completed with zeros (Filling full size field entry is required).
Z	Replaced by any character retrieved from database. Suppress zeros at field left (Complete field filling is optional). When used combined with the mask character X it should be placed at the mask left.

Mask Examples

Mask	Field Value	Formatted Value
(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
(zz) xxxx - xxxx	1234567890	(12) 3456 - 7890
(zz) xxxx - xxxx	12345678	() 1234 - 5678
(zz) xxxx - xxxx	0012345678	() 1234 - 5678

- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.

- **Run content in JavaScript** : If enabled, the JavaScript will be interpreted by the browser, otherwise the JavaScript will be displayed.
- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

PDF Configuration {#id-03}

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

 *Chart layout settings interface.*

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. Integer

Integer

On this page, you will learn how you can configure settings related to the Number field. From the use of specific symbols display to the mode in which they are displayed. And thus, boost the application.

Data type:

It sets the application field type. When the field is set as Number, it is permissible to define formatting rules of integers.

Label:

It sets the title that will be displayed in the field when you run the application. The terminology used in the interface is fundamental for your system has a good usability. In this case, we recommend you to use names and terms familiar to the end user of the application, instead of using terms originated in the system.

Field Mask Input:

It sets the mask input for the field. There are two types of mask as described in the table below:

Character Description

X	It is overridden by any characters returned by the database. Filled is required and the value will be completed with leading zeros when there are fewer characters than the size of the mask.
Z	It is overridden by any characters returned by the database. Its filling is optional and when there are fewer characters than the mask size, nothing will be done about the characters that are missing. In addition the leading zeros will be suppressed. When used in conjunction with the X necessarily the use of this character is to the left of the mask.
9	It represents any numeric character (from 0-9)
A	It represents an alpha numeric character (A-Z,a-z)
*	It represents any alpha-numeric character (A-Z,a-z,0-9)entered by the user.

Mask Input Examples:

Field	Mask Input	Typed Value	Formatted Value
Phone Number	+99 99 9999 - 9999	123456789012	+12 34 5678 - 9012
Phone Number	(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
Phone Number	(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
Phone Number	(zz) xxxx - xxxx	12345678	() 1234 - 5678
Software Key (Letters only)	AAAA-AAAA-AAAA-AAAA	QWERTYUIASDFGHJK	QWER-TYUI-ASDF-GHJK
Software Key (Letters and numbers)	xxxx-xxxx-xxxx-xxxx	Q1W2E3R4T5Y6U7I8	Q1W2-E3R4-T5Y6-U7I8
License Plate	AAA - 9999	QWE1234	QWE - 1234
Scriptcase Serial Key	A999A999A999-**	D111H111G111DG2P	D111H111G111-DG2P
Multiple masks (Phone Number)	9999-9999;(99)9999-9999; 9999 999 9999		+99 99 9999-9999

Repeat value:

This option when enabled will allows you to repeat the field value if it is equal to the value of the previous record in the database.

Example:

Tipo SQL:

Reports the SQL type of the field as it is configured in the database.

Use regional settings:

It allows you to apply regional settings to the number fields formatting. When this option is not selected it will be displayed the attributes group, minus sign and negative number format.

Grouping:

It allows you to define which character is used to separate thousands. This option is only available when the option to use regional settings is disabled.

Negative sign:

It allows you to define which character will be used for the display of negative numbers. This option is only available when the option to use regional settings is disabled.

Negative number format:

It allows you to define the placement of the negative sign in relation to value. This option is only available when the option to use regional settings is disabled.

Color for the negative values:

It allows you to define a color when the value is negative, improving the understanding of the end user about that kind of value.

Example:

Display the value in words:

The value of the field will be displayed in full on application. This feature can facilitate the comprehension and understanding of the user.

Example:

Line size:

Maximum size in characters to be displayed in the value cell, in full. When this value is exceeded the line will break within the cell.

Lookup Settings

The Lookup is a way used to provide the end user a list of values description/name where the items can be selected and used in the application. For example, for a field sex that in the database is represented with M or F values can be presented as male or female dynamically with the lookup. So you will be able to retrieve these values from a database table or manually, if the reference table does not exist.

Example:

Lookup Methods

There are two lookup methods available:

Manual: To use this method it is necessary to inform the selection conditions manually to the Lookup.

Automatic: To use this method it is necessary to use a lookup query from the database to access the reference table.

Editing-automatic lookup:

Select Command:

It sets the SQL command that will retrieve the value to display inside the Grid field. The command must have the following format

```
SELECT field_to_be_displayed FROM table WHERE key_field = {grid_field}
```

The Grid field must be always referenced between curly braces {}, So at run time, the field braces will be replaced by its value in the table.

Multiple Options:

When the Select command informed return multiple values, the option must be selected.

Delimiter:

It sets the tab of values returned by the Select command, this option should be filled when the "Yes" option is checked in the multiple attribute options.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

Use a specific connection:

This option allows you to select another existing connection in the project, other than the current connection, to select the values for the lookup.

Edit Lookup - Manual:

Lookup used to modify the display of the field with predefined values.

Lookup Types

It sets the operating mode of the lookup can be: single value, multiple values (delimiter), multiple values (position) and multiple values (binary).

Single value: it will be displayed only one value for the field. There is a direct replacement of the value stored in the Bank for a label defined in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value of "M" will be replaced by "Male".

Multiple Values (Delimiter):

By using this option it will be displayed multiple values for the selected field. The values must be separated by a delimiter to be informed. A replacement of parts from a string stored in the field, separated by a delimiter for values contained in a list.

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value S;M will be replaced by Sports and Movies.

Delimiter: Character used to separate the values within the same string.

Multiple Values (Position)

Allows you to recover information stored in a single string of the selected field. In order for this information to be retrieved must be informed, in addition to the label, the starting position and the number of bytes that each information occupies within the string.

As an example we will use a string to store Sex, Marital status and Hobby respectively. Sex occupies one byte, Status one byte and Hobby two bytes .

For this purpose we define the list as:

Label	Value	Start	Size
Male	M	1	1
Female	F	1	1
Married	M	2	1
Single	S	2	1
Study	ST	3	2
Sportes	SP	3	2
Reading	LE	3	2

Example: the string MMST would be presented in the query as: Male Married Study

Label: Text that will be displayed in the Grid.

Value: Attribute that matches the value stored in the table. For example, the value M will be replaced by Male.

Start: : Starting position in the string where the information is recorded. The first position is always 1.

Size: Number of bytes that the information occupies in the string.

Multiple values (binary):

It allows you to retrieve several information stored in decimal form of the selected field.

As examples we will use the following list (although not informed in the interface of inclusion of values in the list, each item has a value assigned automatically according to the order):

Assigned value	Description in Lookup
1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

To display the data, a decomposition of the decimal number stored in the bank is performed. For example, numbers 11 and 12 (written in the database) would be decomposed as follows:

$$11 = 1 + 2 + 8 = (\text{Sports} - \text{Culture} - \text{Reading})$$

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

$$12 = 4 + 8 = (\text{Leisure} - \text{Reading})$$

Assigned value Description in Lookup

1	Sports
2	Culture
4	Leisure
8	Reading
16	Music

Label:

Text that will be displayed in the Grid.

Include Button:

Includes in the list the values populated in the Label and value fields.

Alter Button:

It changes the attributes of the selected item.

Delete Button:

Deletes the selected item.

Clean Button:

It cleans the fields

Save Button:

Allows the user to save the entire list, for later use using the Load Lookup definition option.

Load Lookup definition:

Allows you to load a predefined value list for use as a lookup.

Use the lookup in:

Grid: It applies the lookup in all Grid modules (HTML, PDF, XML, CSV, and RTF).

Summary: It applies the lookup only to the Summary (HTML and PDF).

Both This is equivalent to check Grid and Summary options.

Default Value:

Defines a default value, used in cases where the stored value does not match any value from the list.

Display original value and lookup:

When selected **No**, is only displayed the value returned by the Select command. Otherwise, it returns the original value of the field and the value that is returned by the Select command separated by character set in separated by.

Separated By:

Sets the character(s) that will be used to separate the original value of the field and the value that is returned by the Select command. This option should be filled when is checked **Yes** in the Show original value and lookup.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**

- **Font Style** : Allows to select the font family, that will be applied to the application field.
- **Font Size** : Allows to choose the font size, that will be applied to the application field.
- **Italic Text** : Formatting the Text to be Italic.
- **Bold Text** : Formatting the Text to be Bold.
- **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
- **Number of characters** : Amount of characters in bytes to display.
- **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
- **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
- **Background Image** : Background Image of the field.
- **Line Break** : Allows the Line Break.
- **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
- **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
- **Width** : Allows to define a width to the field.
- **Height** : Allows to define a height to the field.
- **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
- **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
- **Bold** : Applies the bold type font to the field.

Line Chart

Using this guide, you can define ac chart for each line.

- **Line Chart Type**: There are two types of charts that can be used. Below you can see their description:
- **Bar Chart** : This option allows to display a bar chart relating to the column that you are working with. The size of the bars is done by math relating to the Grand Total of the column. The value informed in the width of the chart defines the size of the images that will be displayed.

In the example shown above, the width of the Chart is 200 pixels.

- **Line Chart** : This option allows to display a Line Chart relating to the column that you are working with . The amount of lines are done by dividing the value of a column by the value informed to the number of icons.

In the example above, the field Number of Icons, it was informed the value 1000, so the number of stars besides, it determined by dividing the displayed value by 1000.

- **Display Value** : It will display the value and the chart at the same time, the value being on the left or right side of the bar or lines.
- **Number of Icons** : Determines the amount of icons (figures) that will appear for each value on this field. This value will be presented by the dividing the value of the column by the Number of Icons.
- **Chart width** : Width of the chart in pixels. This field is used by the option Bar Chart. In case it is not informed, the value used is 200 pixels.
- **Chart height** : Height of the chart in pixels. This field is used by the option Bar Chart.
- **Margin** : Charts margin. This option is only used by the bar chart.
- **Image / Positive Color** : The way that it is informed will variate depending on the options below:
 - **Bar Chart** : Inform a color for the field that will be used to form the chart image. Click the icon beside to choose the color.
 - **Line Chart** : Inform an image name for the field that will be used like an icon. Click the icon beside to choose the image "Icon".
- **Image / Negative Color** : The information passed to this field follows the same criteria as the Image / Positive Color field. This parameter will be used in the case the value is negative.

Charts Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

 *Bar Chart Interface.*

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

 *Chart layout settings interface.*

1. [Home](#)
2. [Applications](#)
3. Stored Procedures
4. Decimal

Decimal

General Settings

Decimal field Configuration Interface.

- **Data Type** : Select the type of field for the application. When it is defined as a Decimal you can define the format of a decimal number.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Accumulated** : When enabled, define that the field will be accumulated with the values of another field. This option is only available on virtual fields.
- **Accumulator field** : Only available when the option "Accumulated" is enabled. This configuration defines which field will be used to accumulate the values. The chosen fields needs to be Integer, Currency or Decimal.
- **Grid Mask** : Defines the mask for the field display. There are two typed of masks described below:

Character	Description
X	Placeholder to any character. Replaced by any character. If number of characters entered are less then the mask size, the field value is completed with zeros (Filling full size field entry is required).
Z	Replaced by any character retrieved from database. Suppress zeros at field left (Complete field filling is optional). When used combined with the mask character X it should be placed at the mask left.

Mask Examples

Mask	Field Value	Formatted Value
(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
(zz) xxxx - xxxx	1234567890	(12) 3456 - 7890
(zz) xxxx - xxxx	12345678	() 1234 - 5678
(zz) xxxx - xxxx	0012345678	() 1234 - 5678

- **Do Not Repeat Value** : Does not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Values format

Format of Values with Regional Settings.

- **Regional Settings** : Applies the Regional Settings to format the number of the fields. When not enabled, you will view attributes grouped, Negative Symbol and negative number format.
- **Digit Grouping Separator** : Defines the character that will be used to separate the thousand.
- **Decimal Separator** : Defines the character that will be used to separate the decimal.
- **Negative sign** : Defines the character that will be used to display negative numbers.
- **Negative number format** : Defines the position of the Negative sign em relation to the value.
- **Color of Negative** : Defines a color for the field when the value is negative.
- **Decimal Precision** : Number of decimal places. This value is part of the field size.
- **Complete with Zeros** : Defines the decimal places that will or not be completed zeros to the right. Example: The value is 125,50, but the decimal precision is 3. If the option is enabled, then the value will 125,500.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

Attribute Value Lookup Description

1 (Sports) + 4 (Pleasure) + 8 (Reading) =13

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

Line Chart

Using this guide, you can define ac chart for each line.

- **Line Chart Type** : There are two types of charts that can be used. Below you can see their description:
- **Bar Chart** : This option allows to display a bar chart relating to the column that you are working with. The size of the bars is done by math relating to the Grand Total of the column. The value informed in the width of the chart defines the size of the images that will be displayed.

In the example shown above, the width of the Chart is 200 pixels.

- **Line Chart** : This option allows to display a Line Chart relating to the column that you are working with . The amount of lines are done by dividing the value of a column by the value informed to the number of icons.

In the example above, the field Number of Icons, it was informed the value 1000, so the number of stars besides, it determined by dividing the displayed value by 1000.

- **Display Value** : It will display the value and the chart at the same time, the value being on the left or right side of the bar or lines.
- **Number of Icons** : Determines the amount of icons (figures) that will appear for each value on this field. This value will be presented by the dividing the value of the column by the Number of Icons.
- **Chart width** : Width of the chart in pixels. This field is used by the option Bar Chart. In case it is not informed, the value used is 200 pixels.
- **Chart height** : Height of the chart in pixels. This field is used by the option Bar Chart.
- **Margin** : Charts margin. This option is only used by the bar chart.
- **Image / Positive Color** : The way that it is informed will variate depending on the options below:
 - **Bar Chart** : Inform a color for the field that will be used to form the chart image. Click the icon beside to choose the color.
 - **Line Chart** : Inform an image name for the field that will be used like an icon. Click the icon beside to choose the image "Icon".
- **Image / Negative Color** : The information passed to this field follows the same criteria as the Image / Positive Color field. This parameter will be used in the case the value is negative.

```
### PDF Configuration {#id-06}
```

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

 *Chart layout settings interface.*

1. [Home](#)
2. [Applications](#)
3. Stored Procedures
4. Percent

Percent

General Settings

Percentage field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Percentage, it is allowed to define the format of a percentage.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Accumulated** : When enables, define that the field will be accumulated with the values of another field. This option is only available on virtual fields.
- **Accumulator field** : Only available when the option "Accumulated" is enabled. This configuration defines which field will be used to accumulate the values. The chosen fields needs to be Integer, Currency or Decimal.
- **Grid Mask** : Defines the mask for the field display. There are two typed of masks described below:

Character	Description
X	Placeholder to any character. Replaced by any character. If number of characters entered are less then the mask size, the field value is completed with zeros (Filling full size field entry is required).
Z	Replaced by any character retrieved from database. Suppress zeros at field left (Complete field filling is optional). When used combined with the mask character X it should be placed at the mask left.

Mask Examples

Mask	Field Value	Formatted Value
(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
(zz) xxxx - xxxx	1234567890	(12) 3456 - 7890
(zz) xxxx - xxxx	12345678	() 1234 - 5678
(zz) xxxx - xxxx	0012345678	() 1234 - 5678

- **Do Not Repeat Value** : Don't repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Applies the Regional Settings to format the number of the fields. When not enabled, you will view attributes grouped, Negative Symbol and negative number format.
- **Digit Grouping Separator** : Defines the character that will be used to separate the thousand.
- **Decimal Separator** : Defines the character that will be used to separate the decimal.
- **Negative sign** : Defines the character that will be used to display negative numbers.
- **Negative number format** : Defines the position of the Negative sign em relation to the value.
- **Color of Negative** : Defines a color for the field when the value is negative.
- **Decimal Precision** : Number of decimal places. This value is part of the field size.
- **Complete with Zeros** : Defines the decimal places that will or not be completed zeros to the right. Example: The value is 125,50, but the decimal precision is 3. If the option is enabled, then the value will 125,500.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Methods

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

Lookup Method - Automatic

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

Attribute Value Lookup Description

1 (Sports) + 4 (Pleasure) + 8 (Reading) =13

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

Line Chart

Using this guide, you can define ac chart for each line.

- **Line Chart Type** : There are two types of charts that can be used. Below you can see their description:
- **Bar Chart** : This option allows to display a bar chart relating to the column that you are working with. The size of the bars is done by math relating to the Grand Total of the column. The value informed in the width of the chart defines the size of the images that will be displayed.

In the example shown above, the width of the Chart is 200 pixels.

- **Line Chart** : This option allows to display a Line Chart relating to the column that you are working with . The amount of lines are done by dividing the value of a column by the value informed to the number of icons.

In the example above, the field Number of Icons, it was informed the value 1000, so the number of stars besides, it determined by dividing the displayed value by 1000.

- **Display Value** : It will display the value and the chart at the same time, the value being on the left or right side of the bar or lines.
- **Number of Icons** : Determines the amount of icons (figures) that will appear for each value on this field. This value will be presented by the dividing the value of the column by the Number of Icons.
- **Chart width** : Width of the chart in pixels. This field is used by the option Bar Chart. In case it is not informed, the value used is 200 pixels.
- **Chart height** : Height of the chart in pixels. This field is used by the option Bar Chart.
- **Margin** : Charts margin. This option is only used by the bar chart.
- **Image / Positive Color** : The way that it is informed will variate depending on the options below:
 - **Bar Chart** : Inform a color for the field that will be used to form the chart image. Click the icon beside to choose the color.
 - **Line Chart** : Inform an image name for the field that will be used like an icon. Click the icon beside to choose the image "Icon".
- **Image / Negative Color** : The information passed to this field follows the same criteria as the Image / Positive Color field. This parameter will be used in the case the value is negative.

```
### PDF Configuration {#id-06}
```

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

 *Chart layout settings interface.*

1. [Home](#)
2. [Applications](#)
3. Stored Procedures
4. Currency

Currency

General Settings

Currency field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Currency, it is allowed to define the format of a currency.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Accumulated** : When enables, define that the field will be accumulated with the values of another field. This option is only available on virtual fields.
- **Accumulator field** : Only available when the option "Accumulated" is enabled. This configuration defines which field will be used to accumulate the values. The chosen fields needs to be Integer, Currency or Decimal.
- **Grid Mask** : Defines the mask for the field display. There are two typed of masks described below:

Character	Description
X	Placeholder to any character. Replaced by any character. If number of characters entered are less then the mask size, the field value is completed with zeros (Filling full size field entry is required).
Z	Replaced by any character retrieved from database. Suppress zeros at field left (Complete field filling is optional). When used combined with the mask character X it should be placed at the mask left.

Mask Examples

Mask	Field Value	Formatted Value
(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
(zz) xxxx - xxxx	1234567890	(12) 3456 - 7890
(zz) xxxx - xxxx	12345678	() 1234 - 5678
(zz) xxxx - xxxx	0012345678	() 1234 - 5678

- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the datatype of field in the database.

Values Format

Interface of Values Format.

- **Regional Settings** : Allows to apply the Regional Settings to format the number of the fields. When not enabled, you will view attributes grouped, Negative Symbol and negative number format.
- **Currency Format** : Defines the content of the field that if it will be presented with the currency format.
- **Currency Symbol** : Defines the character that represents the Currency Symbol.
- **Digit Grouping Separator** : Defines the character that will be used to separate the thousand.
- **Decimal Separator** : Defines the character that will be used to separate the decimal.
- **Negative sign** : Defines the character that will be used to display negative numbers.
- **Negative number format** : Defines the position of the Negative sign em relation to the value.
- **Color of Negative** : Defines a color for the field when the value is negative.
- **Decimal Precision** : Number of decimal places. This value is part of the field size.
- **Complete with Zeros** : Defines the decimal places that will or not be completed zeros to the right. Example: The value is 125,50, but the decimal precision is 3. If the option is enabled, then the value will 125,500.
- **Display value in full** : The value will be displayed in full. Example: 2018(Two thousand eighteen).

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

- **Lookup Methods**

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

- **Lookup Method - Automatic**

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field

and the value returned from the select command separated by the character in the "Separated by" field.

- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.
- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	

Attribute	Value	Lookup	Description
8		Reading	
16		Music	

1 (Sports) + 4 (Pleasure) + 8 (Reading) = **13**

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
- **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
- **Summary** : Applies the lookup only for the summary (HTML and PDF).
- **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

Line Chart

Using this guide, you can define ac chart for each line.

- **Line Chart Type** : There are two types of charts that can be used. Below you can see their description:
- **Bar Chart** : This option allows to display a bar chart relating to the column that you are working with. The size of the bars is done by math relating to the Grand Total of the column. The value informed in the width of the chart defines the size of the images that will be displayed.

In the example shown above, the width of the Chart is 200 pixels.

- **Line Chart** : This option allows to display a Line Chart relating to the column that you are working with . The amount of lines are done by dividing the value of a column by the value informed to the number of icons.

In the example above, the field Number of Icons, it was informed the value 1000, so the number of stars besides, it determined by dividing the displayed value by 1000.

- **Display Value** : It will display the value and the chart at the same time, the value being on the left or right side of the bar or lines.
- **Number of Icons** : Determines the amount of icons (figures) that will appear for each value on this field. This value will be presented by the dividing the value of the column by the Number of Icons.
- **Chart width** : Width of the chart in pixels. This field is used by the option Bar Chart. In case it is not informed, the value used is 200 pixels.
- **Chart height** : Height of the chart in pixels. This field is used by the option Bar Chart.
- **Margin** : Charts margin. This option is only used by the bar chart.
- **Image / Positive Color** : The way that it is informed will variate depending on the options below:
 - **Bar Chart** : Inform a color for the field that will be used to form the chart image. Click the icon beside to choose the color.
 - **Line Chart** : Inform an image name for the field that will be used like an icon. Click the icon beside to choose the image "Icon".
- **Image / Negative Color** : The information passed to this field follows the same criteria as the Image / Positive Color field. This parameter will be used in the case the value is negative.

```
### PDF Configuration {#id-06}
```

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called

"Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

Chart layout settings interface.

1. [Home](#)
2. [Applications](#)
3. Stored Procedures
4. Percent (Calculated)

Percent (Calculated)

General Settings

Percentage field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Percentage, it is allowed to define the format of a percentage.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Accumulated** : When enables, define that the field will be accumulated with the values of another field. This option is only available on virtual fields.
- **Accumulator field** : Only available when the option "Accumulated" is enabled. This configuration defines which field will be used to accumulate the values. The chosen fields needs to be Integer, Currency or Decimal.
- **Grid Mask** : Defines the mask for the field display. There are two typed of masks described below:

Character	Description
X	Placeholder to any character. Replaced by any character. If number of characters entered are less then the mask size, the field value is completed with zeros (Filling full size field entry is required).
Z	Replaced by any character retrieved from database. Suppress zeros at field left (Complete field filling is optional). When used combined with the mask character X it should be placed at the mask left.

Mask Examples

Mask	Field Value	Formatted Value
(xx) xxxx - xxxx	1234567890	(12) 3456 - 7890
(xx) xxxx - xxxx	12345678	(00) 1234 - 5678
(zz) xxxx - xxxx	1234567890	(12) 3456 - 7890
(zz) xxxx - xxxx	12345678	() 1234 - 5678
(zz) xxxx - xxxx	0012345678	() 1234 - 5678

- **Do Not Repeat Value** : Don't repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

- **Regional Settings** : Applies the Regional Settings to format the number of the fields. When not enabled, you will view attributes grouped, Negative Symbol and negative number format.
- **Digit Grouping Separator** : Defines the character that will be used to separate the thousand.
- **Decimal Separator** : Defines the character that will be used to separate the decimal.
- **Negative sign** : Defines the character that will be used to display negative numbers.
- **Negative number format** : Defines the position of the Negative sign em relation to the value.
- **Color of Negative** : Defines a color for the field when the value is negative.
- **Decimal Precision** : Number of decimal places. This value is part of the field size.
- **Complete with Zeros** : Defines the decimal places that will or not be completed zeros to the right. Example: The value is 125,50, but the decimal precision is 3. If the option is enabled, then the value will 125,500.

Lookup Settings

A method used to offer the user a list of values that can be selected in the form applications. For example, the gender field in the Database with the values M or W can be presented like Man or Women or in a dynamic way, having these values recovered from the database.

Lookup Methods

You will have available two types of lookup methods: Manual (informing manually the conditions of the select) or Automatic (using a database query).

Lookup Method - Automatic

- **SQL Select Statement** : Define the SQL command that will recover the values that will be displayed on the form field. The command needs to have the following structure:

```
SELECT display_field FROM table WHERE key_field = {grid_field}
```

The Grid field needs to be referenced buy the curly brackets {}. While running the application, the field within the curly brackets is replaced by it's value in the table.

- **Multiple Values** : When enabled, allows to select more than one item from the list, separated by a delimiter.
- **Delimiter** : Defines the character used to separate the values selected in the form. Only inform a value to this field if you need to separate the selected options with a different delimiter then (;).
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML, CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
- **Default value** : Defines the default value, used in the cases where the select command does not return any values.
- **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
- **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

- **Choose connection** : Allows you to select another specific connection existing in the project. The select command will be done on the second connection.

Lookup Method - Manual

This method is used so that the developer can create manually the list of values that will be saved to the generated application. This method is mostly used when the database does not have a table that contains this information.

- **Lookup Type** : Define the functionality of the lookup, being: Single Value, Multiple Values (delimiter), Multiple Values (position) and Multiple Values (binary).
- **Single Value** :

It will be displayed only one value for the field. And it will replace the stored value from the table with the one defined on the item label.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Multiple Values (Delimiter)**

It can be displayed various values for the selected field. The values should be separated by a delimiter that will be informed. It is done a replacement on the parts of the field, separated by the delimiter, with the values contained in the list.

- **Label** : Text that will be displayed in the item list of the select field.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Delimiter** Characters used to separate Multiple Values in the table field. If not informed, the default delimiter is a semi coma (;).
- **Multiple Values (position)**

Allows to store a diversity of information selected on the field in only one string. For this information to be added, you need to inform the label, initial position and the number of bytes that each information occupies on a string.

For this example, Gender, Marital status and Hobby were used. Gender uses one byte as does the Marital Status, the Hobby will use two bytes.

The example below:

Label	Value	Start	Size
Man	M	1	1
Women	W	1	1
Divorced	D	2	1
Single	S	2	1
Study	SD	3	2
Sports	SP	3	2
Read	RD	3	2

Example: Of you choose **Man**, **Single** and **Read**, in database would be stored the following value **MSRD**.

- **Label** : Text that will be displayed in the Grid.
- **Value** : Attribute that corresponds to the value that is stored in the table field. For example, the value M will be included into the table when the user will view in the list **Man**.
- **Start** : Starting position of the string that is going to be stored. The first position is always 1.
- **Size** : Amount of bytes that is going to occupy in the string.

Multiple Values (binary)

Allows to store a diversity of information selected on the field in only one value.

In this example uses the following list (although it is not informed when inserting values to the list, each item has a value applied automatically according to the order):

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

To store the data, it is done an internal sum of the decimal number. See the examples below.

Example 1: If the options **Sports** and **Culture** were selected, the number stored in the table would be **3**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

1 (Sports) + 2 (Culture) = **3**

Example 2: If the options **Sports**, **Pleasure** and **Reading** were selected, the number stored in the table would be **13**.

Attribute	Value	Lookup	Description
1		Sports	
2		Culture	
4		Pleasure	
8		Reading	
16		Music	

Attribute Value Lookup Description

1 (Sports) + 4 (Pleasure) + 8 (Reading) =13

- **Label** : Text that will be displayed in the Grid.
- **Insert Button** : Adds to the list the values informed on the fields Label and Value.
- **Update Button** : Modifies the attributes of the selected item.
- **Remove Button** : Remove the selected item from the list.
- **Clear Button** : Clear the Fields.
- **Save Button** : Allows the user to save all the items of the list, to use on other fields, just click on Load lookup definition.
- **Load lookup definitions** : Allows to refresh the list of values predefined when using the lookup. It will be listed the existing definitions in ScriptCase and the ones saved by the user.
- **Use lookup in** :
 - **Grid** : Applies the lookup to all the Grid formats (HTML, PDF, XML , CSV and RTF).
 - **Summary** : Applies the lookup only for the summary (HTML and PDF).
 - **Both** : Applies to both the Grid and Summary.
 - **Default value** : Defines the default value, used in the cases where the select command does not return any values.
 - **Display original and lookup value** : When selected **No**, its displayed on it the value from the select command. Otherwise, it will be displayed the original value of the field and the value returned from the select command separated by the character in the "Separated by" field.
 - **Separated by** : Defines which character will be used to separate the values of the field with the value returned from the lookup. It is only necessary to inform this value of the option Display original and lookup value is enabled.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

Line Chart

Using this guide, you can define ac chart for each line.

- **Line Chart Type** : There are two types of charts that can be used. Below you can see their description:
- **Bar Chart** : This option allows to display a bar chart relating to the column that you are working with. The size of the bars is done by math relating to the Grand Total of the column. The value informed in the width of the chart defines the size of the images that will be displayed.

In the example shown above, the width of the Chart is 200 pixels.

- **Line Chart** : This option allows to display a Line Chart relating to the column that you are working with . The amount of lines are done by dividing the value of a column by the value informed to the number of icons.

In the example above, the field Number of Icons, it was informed the value 1000, so the number of stars besides, it determined by dividing the displayed value by 1000.

- **Display Value** : It will display the value and the chart at the same time, the value being on the left or right side of the bar or lines.
- **Number of Icons** : Determines the amount of icons (figures) that will appear for each value on this field. This value will be presented by the dividing the value of the column by the Number of Icons.
- **Chart width** : Width of the chart in pixels. This field is used by the option Bar Chart. In case it is not informed, the value used is 200 pixels.
- **Chart height** : Height of the chart in pixels. This field is used by the option Bar Chart.
- **Margin** : Charts margin. This option is only used by the bar chart.
- **Image / Positive Color** : The way that it is informed will variate depending on the options below:
 - **Bar Chart** : Inform a color for the field that will be used to form the chart image. Click the icon beside to choose the color.
 - **Line Chart** : Inform an image name for the field that will be used like an icon. Click the icon beside to choose the image "Icon".
- **Image / Negative Color** : The information passed to this field follows the same criteria as the Image / Positive Color field. This parameter will be used in the case the value is negative.

PDF Configuration {#id-06

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

 *Chart layout settings interface.*

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. HTML Image

HTML Image


General Settings

 HTML Image field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as HTML Image, it allows to place an image to the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Image** : Defines an image that will be displayed. The icon "Select Image", lists all the standard images from scriptcase and also the image that you have uploaded to scriptcase. The icon "Upload an image" allows the developer to send an image to the server which is from another machine.
- **Border** : Border size for the image in Pixels.
- **Width** : Define the width of the image in Pixels.
- **Height** : Define the height of the image in Pixels.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

 Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. Credit Card Number

Credit Card Number

General Settings

Credit Card Number Configuration Interface.

- **Data Type** : Select the type of field for the application. When it is defined as a Credit Card Number, you can define some rules for the display format of the Credit Card.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be easier to the user understand if the name would be "Client Name".
- **Do Not Repeat Value** : Do not repeat the value of the field in the case it is the same as the previous record.
- **SQL Type** : Informs the data type of the field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

PDF Configuration {#id-03}

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

Chart layout settings interface.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. E-mail

E-mail

General Settings

Email field Configuration Interface.

- **Data Type** : Defines the type of field for the application. When it is defined as an Email, when you click on the field you be offered a choice for your email client and send an email to that specific email.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

PDF Configuration {#id-03}

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

Chart layout settings interface.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. URL

URL

General Settings

URL field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a URL, it sets the value as a clickable link (Only if it contains a valid path).
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

PDF Configuration {#id-03}

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.
- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.


Chart Layout

Chart layout settings interface.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. YouTube

YouTube


General Settings

 YouTube field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a YouTube, it allows to display a video from youtube on the field.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Display Mode** : Defines how the you video will display in the application.
- **Width** : Width of the video in pixels.
- **Height** : Height of the video in pixels.
- **Link Type** : The display settings of the link, if it is going to be a Text or a Button.
- **Link text** : A text that will link to the video.
- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

 Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. Google Maps

Google Maps

General Settings

Google Maps field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as Google Maps, it will use the Google Maps API to display the map in the Grid Applications
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Home** : Defines what type of parameters will be used in the API.

Home Configuration Interface.

- **Display Mode** : Indicates the display mode of the map. It can be opened in a Modal or in a new Window.
- **Width** : Defines the width of the map that is going to be displayed.
- **Height** : Defines the height of the map that is going to be displayed.
- **Zoom** : Defines the initial Zoom (available from the Google API) of the Map location.
- **API Key** : API Key for authorization to use Google Maps in the Application. (Required only for the versions 2 or earlier of the Google API.)

The API Key is a unique key, composed by a string(text) alphanumeric, which is the license to use the service. When you subscribe to use the service, the key is tied to the domain and the directory of the server. All the pages that use the API needs to be in the same directory that was used for the subscription. In case you have a web server on your local machine, you just need to possess a key for testing, and to do this you only need to set http://localhost in the domain of the subscription.

To get your API Key access the site by [clicking here](#)

- **Link Type** : Defines how the link will be displayed.
- **Text Link** : Text to call the Map.
- **Marker Description** : Displays the description for each marker displayed on the map.
- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. Date

Date

General Settings

Date field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Date and Time, you can define the format of the date.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.

- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

Format of Values without Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator attribute (Check the Regional Settings).
- **Display** : Offers a series of formats predefined for displaying dates.
- **Detail Mask** : Define the format of the field in the Grid, following the standards the PHP function Date.

```
d-m-Y      25-09-2009
F/Y       September/2009
j/n/Y \a\s g:i:s A 25/9/2001 as 14:30:11 PM
l, d \d\e F \d\e Y Thursday, 25 of January of 2009
h:i:s     11:33:20
#h:i:s    123:43:27 (accumulating the
          hours)
```

- **Format Table**

- **Internal Format** : Allows to define the format the field when the SQL type is different from **Date**. You need to use the characters **Y**, **M** and **D** that correspond to **Year**, **Month** and **Day**.
 - **Example 1** : Date stored in a SQL field of the type **char(8)** where the positions **1** to **4** represents the year, the positions **5** and **6** is the month and the position **7** and **8** is the day, define the internal format like: **YYMMDD**
 - **Example 2** : Date stored in a SQL field of the type **char(6)** where the positions **1** and **2** represents the month and the positions **3** to **6** is the year, define the internal format like: **MMYYYY**

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

```
### PDF Configuration {#id-04}
```

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.

- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

 *Chart layout settings interface.*

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. Time

Time

General Settings

Time field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as Time, it is possible to define a Time Format.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

Format of Values without Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator attribute (Check the Regional Settings).
- **Display** : Offers a series of formats predefined for displaying dates.
- **Detail Mask** : Define the format of the field in the Grid, following the standards the PHP function Date.

```
d-m-Y      25-09-2009
F/Y       September/2009
j/n/Y \a\s g:i:s A 25/9/2001 as 14:30:11 PM
l, d \d\e F \d\e Y      Thursday, 25 of January of 2009
h:i:s     11:33:20
#h:i:s    123:43:27 (accumulating the
           hours)
```

- **Format Table**

- **Internal Format** : Allows to define the format the field when the SQL type is different from **Date**. You need to use the characters **A**, **M** and **D** that correspond to **Year**, **Month** and **Day**.
 - **Example 1** : Date stored in a SQL field of the type **char(8)** where the positions **1** to **4** represents the year, the positions **5** and **6** is the month and the position **7** and **8** is the day, define the internal format like: **YYMMDD**
 - **Example 2** : Date stored in a SQL field of the type **char(6)** where the positions **1** and **2** represents the month and the positions **3** to **6** is the year, define the internal format like: **MMYYYY**

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

```
### PDF Configuration {#id-04}
```

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.

- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.

Chart Layout

 *Chart layout settings interface.*

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. Date and time

Date and time

General Settings

Date and Time field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Time, you can define the format of the time.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as `cmp_name_client`, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Do Not Repeat Value** : Do not repeat the value of the field in case it is the same as the previous record.
- **SQL Type** : Informs the data type of field in the database.

Values Format

Format of Values with Regional Settings.

Format of Values without Regional Settings.

- **Regional Settings** : Allows to apply the Regional Settings to format the dates of the field. When not enabled, it will be displayed the date separator attribute (Check the Regional Settings).
- **Display** : Offers a series of formats predefined for displaying dates.
- **Detail Mask** : Define the format of the field in the Grid, following the standards the PHP function Date.

```
d-m-Y      25-09-2009
F/Y        September/2009
j/n/Y \a\s g:i:s A 25/9/2001 as 14:30:11 PM
l, d \d\e F \d\e Y      Thursday, 25 of January of 2009
h:i:s      11:33:20
#h:i:s     123:43:27 (accumulating the
             hours)
```

- **Format Table**

- **Internal Format** : Allows to define the format the field when the SQL type is different from **Date**. You need to use the characters **Y**, **M** and **D** that correspond to **Year**, **Month** and **Day**.
 - **Example 1** : Date stored in a SQL field of the type **char(8)** where the positions **1** to **4** represents the year, the positions **5** and **6** is the month and the position **7** and **8** is the day, define the internal format like: **YYMMDD**
 - **Example 2** : Date stored in a SQL field of the type **char(6)** where the positions **1** and **2** represents the month and the positions **3** to **6** is the year, define the internal format like: **MMYYYY**

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

```
### PDF Configuration {#id-04}
```

Chart Settings

By using this option, you will see an icon on the Field Title, this icon displays a chart relating to the field.

On this type option you need to choose two columns to build the chart. The first one is the field its self that you are working with, and the second one is the you chose and is called "Column for Label".

Bar Chart Interface.

- **Group by Label** : Groups the values of the column by the label. Similar to the effect of the group by of the a Select.
- **Summary Function** : Function that summarizes the data applied to the column. The functions that can be used on the field are the following: Count, Sum, Max, Min and Avg.

- **Configurable Chart** : Allows the user to setup the parameters of the chart creation when the application is running.
- **Display Values** : Displays the values of the generated chart.
- **Column for Label** : Selecting the column that will be the label for the field.
- **Chart Width** : Width of the chart, in pixels.
- **Chart Height** : Height of the chart, in pixels.


Chart Layout

 *Chart layout settings interface.*

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. Image (Database)

Image (Database)


General Settings

 Upload field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as an Image (Database), all the Images files are stored and loaded directly from the Database.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Image Border** : Width of the Image border. Value in Pixels.
- **Image Height** : Image height size. Value in Pixels.
- **Image Width** : Image width size. Value in Pixels.
- **Maintain Aspect** : Maintains the original aspect ratio of the image when resizing it.
- **Open in Another Window** : Allows to open the image in another window.
- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

 Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. Image (File Name)

Image (File Name)

General Settings

Upload field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as an Image (File Name), all the images files are stored and loaded in a directory of the server (Only the image name is stored in the database).
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Image Border** : Width of the Image border. Value in Pixels.
- **Image Height** : Image height size. Value in Pixels.
- **Image Width** : Image width size. Value in Pixels.
- **Maintain Aspect** : Maintains the original aspect ratio of the image when resizing it.
- **Open in Another Window** : Allows to open the image in another window.
- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.


Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. Document (Database)

Document (Database)


General Settings

 Upload field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Document (Database), all the document files are stored and loaded directly from the Database.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Icon** : Displays an icon beside the field to identify the type of document.
- **File Name** : Defines the field that contains the name of the document stored in the database (the field that contains this information also needs to be stored in the database).
- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

 Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. Document (File Name)

Document (File Name)

General Settings

Upload field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a Document (File Name), all the document files are stored and loaded in a directory of the server (Only the document name is stored in the database).
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **Sub-folder** : Sub-folder name that the files are stored.
- **Icon** : Displays an icon beside the field to identify the type of document.
- **File Name** : Defines the field that contains the name of the document stored in the server (the field that contains this information also needs to be stored in the database).
- **SQL Type** : Informs the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. Bar code

Bar code

General Settings

 Configuration Interface of the Barcode Field.

- **Data Type** : DataType of the field for the application.
- **Label** : Set the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name would be "Client Name".
- **SQL Type** : Database field type.

Watch below a video showing an example about the Barcode field:



Values Format

 Configuration Interface of the Barcode Field.

- **Type** : Type of Barcode.
- **Text** : Barcode Text for illustration purposes.
- There are **18 types of barcodes** that are listed below:

 Barcode configuration interface.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.


 Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. QRCode

QRCode

General Settings

 QRCODE field Configuration Interface.

- **Data Type** : Define the type of field for the application. When it is defined as a QRCODE, allows you to set values into a QRCODE.
- **Label** : Defines the title of a field in the application. For example: if the field is defined in the database as cmp_name_client, it will be more clear for the comprehension of the user if the name would be "Client Name".
- **SQL Type** : Informs the data type of field in the database.

Watch below a video showing an example with QRCODE



Values Format


 Interface of Values Format.

- **Level of error correction**: Codewords are 8 bits long and use the Reed-Solomon error correction algorithm with four error correction levels. The higher the error correction level, the less storage capacity.
- **Image Size** : Size of the QRCODE.
- **Margin** : Margin of the QRCODE.

 Interface of Values Format.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of scriptcase, there are the same attributes available in Interface.

 Display Settings configuration Interface.

- **Field Settings**
 - **Font Style** : Allows to select the font family, that will be applied to the application field.
 - **Font Size** : Allows to choose the font size, that will be applied to the application field.
 - **Italic Text** : Formatting the Text to be Italic.
 - **Bold Text** : Formatting the Text to be Bold.
 - **Don't truncate** : Do not truncate by the amount of bytes or by the Grid columns.
 - **Number of characters** : Amount of characters in bytes to display.
 - **Horizontal Alignment** : Allows to position the label of the filter in the desired location (left right, center and justify).
 - **Vertical Alignment** : Allows to position the label of the filter in the desired location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Background Image** : Background Image of the field.
 - **Line Break** : Allows the Line Break.
 - **Font Color** : Allows to apply a color to the font by choosing a color from the color palette.
 - **Background Color** : Allows to apply a color to the field background by choosing a color from the color palette.
 - **Width** : Allows to define a width to the field.
 - **Height** : Allows to define a height to the field.
 - **Title Horizontal Alignment** : Define the Title Horizontal Alignment of the field.
 - **Title Vertical Alignment** : Define the Title Vertical Alignment of the field.
 - **Bold** : Applies the bold type font to the field.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. Signature

Signature

General Settings

Configuration Interface of the Signature Field.

The signature field will help you creating more sophisticated forms and making it possible to store signatures in your database. Inside our development environment we have specific settings that will help you to customize your field, those options are:

- **Data Type** : You can define the type of field for the application. When it is defined as a text, it accepts letters, numbers and special characters.
- **Label** : Lets you define a label to the field in the application. For example: if the field is defined in the database as cmp_name_client, it will be easier to the user understand if the name is "Client Name".
- **Background color** : Defines a color to the field background by using a valid hexadecimal color value or by choosing a color from the color palette.
- **Pen color**: Set a color to the pen by using a valid hexadecimal color value or by choosing a color from the color palette.
- **Width** : Lets you define a width to the field.
- **Height** : Set a height to the field.
- **Subtitle** : Defines the subtitle that will be displayed beside the field.
- **Initial Value** : Lets you define the initial value for the field when in insert mode. The options are:
 - **Defined Value** : The field will receive the value from the text field of the defined value option.
- **Save Variable** : Save a variable in the session with the value of the field, that can be used in other applications. For example, in the login form the user name can be saved in the session so that it can be displayed on the header of other applications.
- **Variable Name** : Set the name for the session variable that will receive the field value.
- **Disabled Field** : Define if the field will be disabled in "Insert Mode", "Update Mode" or in "Insert and Update Mode".
- **HTML Type** : Displays the HTML Object that will be used to display the field in the form.
- **SQL Type** : Displays the data type of field in the database.

Display Settings

The Display Settings allows to define the CSS values individually for each field. For each Display schema of ScriptCase, there are the same attributes available in Interface.

Display Settings configuration Interface.

- **CSS of the Title**
 - **Font** : Select the font type, that will be applied to the application field title.
 - **Font Size** : Defines the font size, that will be applied to the application field title.
 - **Font Color** : Choose a color to the font by using a valid hexadecimal color value or from the color picker.
 - **Background Color** : You can define the color for the field by using a valid hexadecimal color value or from the color picker.
 - **Bold** : Applies the bold style to the font.
 - **Underline** : Set the underline style to the font.
 - **Border style** : Choose a style for the border.
 - **Collapse** : Defines the collapse for the border.
 - **Border Size** : Applies the border size to the title of the field.
 - **Border Color** : Choose a color for the border, using a color palette to apply to the title.
 - **Horizontal Alignment** : Position the label of the field in the wanted position (left,right,center and justify).
 - **Vertical Alignment** : Position the label of the field in the wanted position (baseline, sub, super, top, text-top, middle, bottom, text-bottom).
 - **Width** : Defines a width for the title of the field.
 - **Height** : Set a height for the title of the field.

□

1. [Home](#)
2. [Applications](#)
3. Stored Procedures
4. Rating

Rating

This field allows the developer to create a rate field to display the data previously added using stars (or any other image).

Description of the attributes available for configuration.

Data type

Define the type of the field in the application. In this case we must select the type **Rating**.

Label

Define the application title that will be showed when the application is executed. The terminology used in the interface of your application is very important for the usability, and we need to use familiar terms for the final user, instead of use terms from the system.

Example, this field "rating" that is named **cmp_rating** however, the customer would have a better understanding if the label uses **Product Rating**.

Besides use a fixed text, the **Label** attribute allows the use of **Langs** to define the field title, allowing the [internationalization of your application](#).

Subtitle

Define the subtitle that will be shown in the field below the ratings. **Example**: "Thank you for your feedback!".

As in the **Label**, the **subtitle** attribute also allows the use of **Langs** for [internationalization of your application](#).

Amount of icons

Define the number of icons that will be displayed in the field. The value set in this attribute must be according to the way the evaluation will be realized.

To evaluate using Stars normally it is used five stars, so , we must to inform 5 in this attribute. After we set the images that will be displayed in the attribute **Enabled icon** and **Disabled icon** the result will be the same as in the example:

To use Like/Dislike **for example** we must inform 1 in the **amount of icons**, so only one image will be displayed and will change according to the click. In this case, the storage in the database will be 1 for like (enabled icon) and 0 for dislike (disabled icon). **You can see the field configured below as like/dislike:**

Enabled icon

Define the icon that will be shown for the amount selected. Following the like/dislike example, in this option, we must insert the Like image . We must do the same for a star evaluation, where for default we must insert the stars filled .

Disabled icon

Define the icon that will be shown for the not selected icons. In this attribute, we set an image to represent the opposite of the previous images set in the **Enabled icon**.

Still referring the example of the Like/Dislike, in this attribute we must set the Dislike .

As in the stars example, we must insert the image with the empty stars .

SQL Type

Inform the type of the field in the database.

Configuração da Visualização

You can set CSS values for the field individually. Thus, for example, you can highlight one field of the others in the application. As CSS field properties, when changed, they are added to a class created automatically by ScriptCase for each application field.

Individual field CSS settings, when inserted, override theme settings ([CSS of applications \(Themes\)](#)) selected for application.

The settings are divided into three property blocks, these blocks are:

CSS of Title

This makes it possible to change the CSS properties of the field's Label.

In the example below, You can see the difference of field title configurations. While the fields **provider** and **Category** have the same formatting, inherits the theme of the application, the field **product evaluation** have a different formation of of others, from the changes made in the CSS of the field.

CSS of field

Changes the CSS properties of the <td> where the input object (where the user enters data for insertion into a form) is positioned. In the image below, you can see where the change is applied.

Field with changes to Field CSS properties, changing background color and horizontal alignment

CSS of Input object

Changes the CSS properties in the Input of field object, where the user type the data in a form.

Field with changes to CSS properties of Input Object, changing background color and input source color**CSS properties**

The available configuration options are basically the same for each of the configuration blocks above.

description of available configuration attributes**Source**

Changes the font of the text according to the fonts selected using the *font-family* property.

In this option, some types of fonts are provided to you (as shown below).

Size

Changes the font size of the text using the property *font-size* in the field class.

You need to select the available value from our list, the measure used for this property is the pixel.

Font Color

Define the font color used by the property *color* in the field class.

Background color

Define the background color using the property *background-color* in the field class.

The colors that will be used in the two color properties listed above, **Font color** and **Background color**, can be entered via the color palette - ![Color palette icon][palette] - available next to the field or manually entered values in the accepted formats that are: *Hexadecimal, RGB, RGBA, HSL, HSLA* or *Color name*.

Color palette

By clicking on the color palette icon - ![Color palette icon] [palette] - next to the field, a window will open with some default colors.

When you select one of the colors, a value in hexadecimal format (HEX) will be entered, representing the chosen color.

Hexadecimal

Acronym for hexadecimal, this code is composed of the pound sign (#) plus six digits. The first two define the intensity of the color red, the middle two are green and the last two are blue.

This code is represented by #000000.

In this representation system #000000 represents the color black and #ffff represents the color white.

RGB / RGBA Acronym for Red, Green and Blue. This code is represented by RGB (255,255,255).

The first three numbers defined with intensity of red color, the middle three numbers are green and the last three numbers are blue.

HSL / HSLA**Color name**

You need to enter the name of the color you want to use. This name must be entered in English and the color must be within the list of browser supported colors you will use.

Example:

Some sites provide a color list with each of the above values.

This property allows you to specify colors in different ways using different ways of value assignment

You can also enter values directly in the field without using our colors palette. This property lets you specify as colors different ways to assign values to CSS properties that support colors.

Bold

Applies bold style to the font.

Underline

Lets you apply the underlined style to the font.

Border style

Defines the border font style.

Border Collapse

Defines the border collapse.

Border size

Changes the size of the title border.

Border color

Chooses the border color , using a color palette to apply to the title.

Horizontal alignment

Position the filter label at desired location (left, right, center e justify).

Vertical alignment

Position the label of the filter in the wanted location (baseline, sub, super, top, text-top, middle, bottom, text-bottom).

Not available in Input Object CSS

Width

To set the width of the title.

Height

To set the height of the title.

Allow the developer to set up instructions and hints about the field's use, allowing the developer to help the users about the system usage.

Description of the settings attributes.**Help Description**

In this area, you can set up a text that will be shown in the application according to the type of help selected below.

Help Type

Pop-up - Display a default icon ! beside the field. To display the message just click in the icon that a Pop-up will be displayed with the help text.

Hint - Display a default icon ! beside the field. When the mouse is above the icon a help text will be displayed.

The icon displayed when selecting the types **Hint** or **Pop-up** vary according to the theme that is being used in the application. This icon can be modified in the CSS Buttons.

Text - Display a text inside of a `` beside the field.

Help button position

Allow to set up where the help button will be displayed in the field, there are two options:

Beside: The help icon will be displayed beside the field.

Column: The help icon will be displayed beside the label of the field.

Blocks

Blocks are “containers” where you can position the application fields of Forms, Controls, or Slide Grids.

Scriptcase creates applications with one block by default. You can add more blocks as you wish, to organize it in the best way.

See below, the Columns Organization, and where you can define the position of the next block: beside or below the current one.

		Block	Title	Label		Fields		Organization				
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse		
Pag1												
			form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Pag2												
			Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Blocks not shown												

Application Block configuration

On the left side of each block, there are two icons, the first one to edit the information of the block and the second one to delete the block.

Organizing The Position Of The Blocks

See below how to modify the display order of the Blocks in one Page.

Click and drag the block that you desire to modify to its new position.

		Block	Title	Label		Fields		Organization				
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse		
Pag1												
			form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Pag2												
			Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Blocks not shown												

Application Block Display configuration

- see how to remove a block from the display

Click on the block desired and drag it to the item “Blocks not Shown”. This way, you can also drag the block to another page if desired. See the images below.

		Block	Title	Label		Fields		Organization				
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse		
Pag1												
			form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Pag2												
			Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Blocks not shown												
			Block 3	Block 3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼

Application Block Display configuration

		Block	Title	Label		Fields		Organization				
Op	Name	Label	Display	Display	Position	Columns	Position	Next	Width	Collapse		
Pag1												
			form_orders	form_orders	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Pag2												
			Block 2	Block 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼
Blocks not shown												
			Block 3	Block 3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Beside ▼	1	Beside ▼	Below ▼	100%	Start open ▼

Application Block Display configuration

Block

- **ame:** The name that identifier the Block.
- **abel:** Title of the block to display in the application.

Title

- **isplay:** It controls the display of the block title.

Label

- **isplay:** It controls the display of the field labels of the block.
- **osition:** Options to display label :
 1. **bove:** Display the label above the field.
 2. **eside:** Display the label beside the field.
 3. **elow:** Display the label below the field.

Fields

- **olumns:** Amount of columns side by side in the block.
- **osition:** The way to display the fields in the block:
 1. **elow:** Display the fields one below the other respecting the number of columns.
 2. **eside:** Display the fields one beside the other respecting the number of columns.
 3. **ine:** Display the fields one beside the other with no tabulation.

Organization

- **ext:** The way to display the blocks in the page:
 1. **elow:** Set to show the following block below the current one.
 2. **eside:** Set to show the following block beside the current one.
 3. **abs:** Set to show the following block in a different tab then the current one.
- **idth:** Set the block width in pixels or percentages. Use the symbol "%" to indicates the value in percentage.
- **ollapse:** Enables the option to close the block.

Create A New Block

To include new blocks in an Application, click on the button [Create New Block](#). Then, enter the name and label of the block in the following interface and finish by click on Create.

Creating application blocks configuration


Name






Name of the Block.

Label

Title of the block to display in the application.

Edit Blocks

To edit a block, click on the icon , that is on the left side of the block. Then you can see the following interface to define the parameters of the blocks. Click on Save to finish.

EDIT BLOCKS	
ATTRIBUTE	VALUE
Name	<input type="text" value="form_orders"/>
Title	<input type="text" value="form_orders"/>
Display Title	<input type="radio"/> Yes <input checked="" type="radio"/> No
Title Font	<input type="text" value=""/> Aa
Font Size	<input type="text" value=""/>
Font Color	<input type="text" value=""/> 
Background Color	<input type="text" value=""/> 
Background image	<input type="text" value=""/> 
Title Height	<input type="text" value="20"/> pixels
Horizontal Alignment	<input type="text" value=""/>
Vertical Alignment	<input type="text" value=""/>
Display Label	<input checked="" type="radio"/> Yes <input type="radio"/> No
Columns	<input type="text" value="1"/>
Columns Width	<input type="text" value="Calculated"/>
Label Color	<input type="text" value=""/> 
Fields Organization	<input type="text" value="Beside"/>
Label Position	<input type="text" value="Beside"/>
Next Block	<input type="text" value="Below"/>
Border Color	<input type="text" value=""/> 
Border Width	<input type="text" value="0"/> pixels
Block Width	<input type="text" value="100%"/>
Block Height	<input type="text" value=""/>
Cell Spacing	<input type="text" value=""/> pixels
Collapse	<input type="text" value="Start open"/>

Application Block editing interface

Name

Name of the block. ##### Title
 Block title for display. ##### Display Title
 This option, when active, allows displaying the block title. ##### Title Font
 Set the font family of the block title. ##### Font Size
 Set the font size of the block title. ##### Font Color
 Set the font color of the block title. ##### Background Color
 Set the Background Color of the block title. ##### Background image
 Set a Background image for the block title. ##### Title Height
 Height in pixels of the block title line. ##### Horizontal Alignment
 Horizontal Alignment of the block title (Left, Center, and Right). ##### Vertical Alignment
 Vertical Alignment of the block title (Top, Middle, and Bottom). ##### Display Label
 Display the labels of the fields in the block. ##### Columns
 Amount of field columns in a block. ##### Columns Width
 Set the field column width of the block. ##### Label Color
 Color of the field labels. ##### Fields Organization
 The way to display the fields in the block. ##### Label Position
 Set the position of the field labels of the block. ##### Next Block
 Set the position of the following block relating to the current one. ##### Border Color
 The border Color for the block. ##### Border Width
 The border Width for the block. ##### Block Width
 The width for the block. ##### Block Height
 The Height for the block. ##### Cell Spacing
 The Cell Spacing in the block. ##### Collapse
 It enables the option to close the block.

This module lets you editing display options for the application. You can apply visualization themes, organize blocks, define values, and the display format of the Header and Footer.

Layout Settings

ScriptCase creates an app by using default values defined per project, but you can change those values for a specific application on this interface.

▾ LAYOUT SETTINGS

Header Template	<input type="text" value="Flat"/>	Template name used for the application header.
Footer Template	<input type="text" value="Default"/>	Template name used for the application footer.
Button	<input type="text"/>	Use different buttons than what was defined in the color scheme.
Themes	<input type="text" value="Sc9_Rhino"/>	Use different themes from the one defined for the color scheme

Header

|< < > >| xxyyzz xxxxx yyyyy ▾

Block 1

Name

Type Male Female

Address*

Groups* Male Female

Countries ▾

Address

Drag & Drop files here

Photos

Image1.png ✓

Image2.png ✗

Captcha

Application Layout Settings

Header Template

Allows choosing the look of the application Header.

Footer Template

Allows choosing the look of the application Footer.

Button

Allows choosing the button theme for the application.

Themes

Choose one of the selected themes in the project properties. It defines the look of the application, like colors, fonts, and others.

Header & Footer

Header

Here we can define if we want to display the header or not, the application title, and fill the header variables. The header looks according to the selected template in the Layout settings interface.

HEADER	
ATTRIBUTE	VALUE
Display Header	<input checked="" type="checkbox"/>
Insertion Title	{lang_othr_frmi_titl} - orders
Update Title	{lang_othr_frmu_titl} - orders
Header Variables	
SC_HEADER	Title
SC_VALUE	Date

Application Header Configuration

Display Header


This option determines if we want to display the header or not.

Title

Allows informing the title of the application.

Header Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **itle**: It displays the value of "**Application Title**" in the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Depending on the Application, you may have more than one title option.

Footer

FOOTER	
ATTRIBUTE	VALUE
Display Footer	<input type="checkbox"/>
Footer Variables	
NM_FOOTER_DATA	
NM_FOOTER_LOGO	
NM_FOOTER_TITLE	

Application Footer Settings


This page may change depending on the footer template chosen in the Layout Settings.

Display Footer

This option determines if we want to display the footer or not.

Footer Variables

We must fill the variable fields with one of the types available in the Combobox. Depending on the type, you need to associate content with it. Those are the types:

- **ield**: When you choose the option "**Field**", it opens a Combobox to choose a field to associate the field value with the header.
- **ate**: It displays the system's date in "yyyy/mm/dd" format in the header. There are several display formats available, and you can change it by using the text field beside it. To access existing formats, click on the hint to see an explanation.
- **mage**: It displays a field to inform the name of an existing image in the server. You can select an image by clicking on the icon "**Choose Image**", and you still can upload new images by using the button "**Upload**". 
- **alue**: It displays the content of the text input. You can inform static texts and "**Global Variables**". e.g. "Employee Name: [v_name]".

Scriptcase has Events to allow the developers to customize the application code. By using the events, the developers can program custom actions at a specific execution time (e.g. After a record is inserted, when it is loaded, upon submitting a Form, ...) and for a specific application type. In the events area you can use global and local variables, JavaScript, CSS, HTML, PHP codes and also Scriptcase macros.

onApplicationInit

This event occurs only once, when the application is loaded and before the application executes the SQL. It can be used for data processing or checking variables.

Example: `if ([glo_var_dept] != 'financial'){ sc_redir(app_x.php); }`

onNavigate

This event occurs when navigating between the application pages.

OnScriptInit

This event occurs when the application scripts are started. It runs before the OnLoad events(when present), however, it will always be executed when running, reloading or navigating the application.

onRecord

This event is executed at the moment the record line is loading. The OnRecord event is a loop that loads the records by each row.

onHeader

This event runs when the header of the application is loaded.

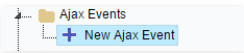
onFooter

This event runs when the footer of the application is loaded.

OnClick

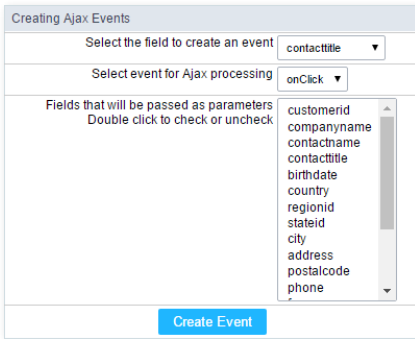
The ajax event OnClick is executed when the field that it's based on is clicked.

- creating a new ajax event



Creating a new ajax event

- selecting a field



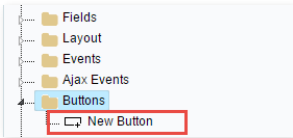
Creating Ajax Events	
Select the field to create an event	contacttitle
Select event for Ajax processing	onClick
Fields that will be passed as parameters Double click to check or uncheck	customerid companyname contactname contacttitle birthdate country regionid stateid city address postalcode phone
Create Event	

Selecting a field of the ajax event

Choose a field to create an event To define in which field the event will be added to.

Choose an event that runs the ajax Defines which event will be added to the field.

In addition to the buttons that comes automatically with the applications, you can also create your own buttons. All the buttons are displayed within the application toolbar.

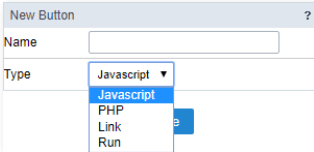


New buttons creation settings

Creating a new button

To create a new button, click on the “New Button” option and enter a name and a button type.

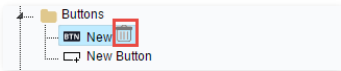
The button Types are: JavaScript, PHP, Link and Run.



Grid buttons type

Deleting a button

To delete a button click on Delete icon (represented by a recycle bin image) in the right of the button name, at the application menu under the Buttons option.



Deleting a button

JavaScript

Display Mode

You can configure the display mode of the javascript button in Image, Button or Link.

Button

ATTRIBUTE	VALUE
Display Mode	Button
Label	JavaScript
Hint	
Confirmation Message	
Type	JavaScript

Setting up Javascript Button.

Display Mode You can select the display mode for the javascript button in this option.

Label Text of the button that will be display in the application while executing.

Hint Hint message for the button. (Displayed when the mouse hovers the button).

Confirmation Message Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.

Type Description of the created button.

Image

ATTRIBUTE	VALUE
Display Mode	Image
Icon	
Hint	
Confirmation Message	
Type	JavaScript

Setting up Javascript Button.

Display Mode You can select the display mode for the javascript button in this option.

Icon Allows you to inform the icon that will be displayed on the button while the execution of the application.

Hint Hint message for the button. (Displayed when the mouse hovers the button).

Confirmation Message Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.

Type Description of the created button.

Link

ATTRIBUTE	VALUE
Display Mode	Link
Label	JavaScript
Hint	
Confirmation Message	
CSS Style	default
Type	JavaScript

Setting up Javascript Button.

Display Mode You can select the display mode for the ajax button in this option.

Label Text of the button that will be display in the application while executing.

Hint Hint message for the button. (Displayed when the mouse hovers the button).

Confirmation Message Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.

CSS Style Name of the CSS class, style created in the layout editor.

Type Description of the created button.

Code Block

Javascript button coding block.

In this block, only JavaScript is accepted.

PHP

Display Mode

You can configure the display mode of the PHP button in Image, Button or Link.

Button

ATTRIBUTE	VALUE
Display Mode	Button
Label	PHP
Hint	
Confirmation Message	
Reload quantity of records	No
Type	PHP
Target	Same Window

Setting up PHP Button.

Display Mode You can select the display mode for the link button in this option.

Label Text of the button that will be display in the application while executing.

Hint Hint message for the button. (Displayed when the mouse hovers the button).

Confirmation Message Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.

Reload quantity of records This option is used to update the amount of records in the application.

Type Description of the created button.

Target Defines the window destination where the code will run (Open in the same Window, Open in another Window, Modal).

Image

ATTRIBUTE	VALUE
Display Mode	Image ▼
Icon	<input type="text"/>
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
Reload quantity of records	No ▼
Type	PHP
Target	Same Window ▼

Setting up PHP Button.

- Display Mode** You can select the display mode for the link button in this option.
- Icon** Allows you to inform the icon that will be displayed on the button while the execution of the application.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- Reload quantity of records** This option is used to update the amount of records in the application.
- Type** Description of the created button.
- Target** Defines the window destination where the code will run (Open in the same Window, Open in another Window, Modal).

Link

ATTRIBUTE	VALUE
Display Mode	Link ▼
Label	PHP
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
CSS Style	default ▼
Reload quantity of records	No ▼
Type	PHP
Target	Same Window ▼

Setting up PHP Button.

- Display Mode** You can select the display mode for the link button in this option.
- Label** Text of the button that will be display in the application while executing.
- Hint** Hint message for the button. (Displayed when the mouse hovers the button).
- Confirmation Message** Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
- CSS Style** Name of the CSS class, style created in the layout editor.
- Reload quantity of records** This option is used to update the amount of records in the application.
- Type** Description of the created button.
- Target** Defines the window destination where the code will run (Open in the same Window, Open in another Window, Modal).

Code Block

ATTRIBUTE	VALUE
Code	*Ajax button coding block. *

In this block, you can use macros, PHP code and JavaScript.

Link Button

Display Mode

You can configure the display mode of the link button in Image, Button or Link.

Button


ATTRIBUTE	VALUE
Display Mode	Button ▼
Label	Link
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
Type	Link

Setting up Link Button.

Display Mode	You can select the display mode for the link button in this option.
Label	Text of the button that will be display in the application while executing.
Hint	Hint message for the button. (Displayed when the mouse hovers the button).
Confirmation Message	Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
Type	Description of the created button.

Image

► Button Settings: Link

ATTRIBUTE	VALUE
Display Mode	Image ▾
Icon	<input type="text"/> 
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
Type	Link

[Link](#)

Setting up Link Button.

Display Mode	You can select the display mode for the link button in this option.
Icon	Allows you to inform the icon that will be displayed on the button while the execution of the application.
Hint	Hint message for the button. (Displayed when the mouse hovers the button).
Confirmation Message	Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
Type	Description of the created button.

Link

► Button Settings: Link

ATTRIBUTE	VALUE
Display Mode	Link ▾
Label	Link <input type="text"/>
Hint	<input type="text"/>
Confirmation Message	<input type="text"/>
CSS Style	default ▾
Type	Link

[Link](#)

Setting up link Button.

Display Mode	You can select the display mode for the ajax button in this option.
Label	Text of the button that will be display in the application while executing.
Hint	Hint message for the button. (Displayed when the mouse hovers the button).
Confirmation Message	Displays a confirmation message when the button is clicked. If not informed this is not informed, no message will appear.
CSS Style	Name of the CSS class, style created in the layout editor.
Type	Description of the created button.

Setting Up The Link

- electing the applications

Link between applications - Application List

Select the application that will be called:

Application All By folder By type

- calendar_events
- chart_customers
- dashboard
- form_customers
- form_employees
- form_orders
- form_sec_users
- grid_categories

[◀ Back](#) [Next ▶](#) [Help](#)

Choosing the application for the button link.

You should select an application to be called from the button link.

- ink Parameters

Choosing the parameters for the button link.

- Field** Allows you to use an existing field from the current application as a parameter for the link.
- Variable** Allows you to use a global variable from the current application as a parameter for the link.
- Fixed** Allows you to inform a fixed value as a parameter for the link.
- Empty** No value will be passed as a parameter for the link.
 - ink Properties (Grid)

Configuring the properties for the link button when the destined application is a Grid.

- Link Operation Mode** How the link will open (Open in the same Window, Open in another Window, Modal).
- Exit URL for the target application** URL or an application that be redirect to when exiting the Grid application.
- Initial Mode** Allows you to define the initial mode of the grid application (Search or Grid).
- Number of Lines** Allows you to define the amount of lines displayed in the Grid.
- Number of Columns** Allows you to define the amount of columns displayed in the Grid.
- Paging** Enable the paging in the Grid.
- Display Header** Enable the Grid Header.
- Active Navigation Buttons** Enable the navigation button (First, Back, Next and Last) in the Grid.
 - ink Properties (Form)

Configuring the properties for the link button when the destined application is a Form.

- Link Operation Mode** How the link will open (Open in the same Window, Open in another Window, Modal).
- Exit URL for the target application** URL or an application that be redirect to when exiting the Form application.
- Enable insert button on target application** Enable the "New" button in the Form Application.
- Enable update button on target application** Enable the "Update" button in the Form Application.
- Enable delete button on target application** Enable the "Delete" button in the Form Application.
- Enable navigation button on target application** Enable the navigation button (First, Back, Next and Last) in the Form.
- Enable button to edit a grid record** Enable the buttons that allow you to edit the records of a Grid

RUN

Display options

We can configure Run Button display as Link, Image or Button:


Button

ATTRIBUTE	VALUE
Display Mode	Button
Label	Run
Hint	
Confirmation Message	
Reload quantity of records	No
Type	Run
Target	Same Window

Grid's Run Button settings - Button type

- Display Mode** Combo box to select the button display option, you can choose button, image or link.
- Label** Application button title (text that will be displayed for the button within the application)
- Hint** Using this option you can set a message for the button hint
- Confirmation Message** Using this option you can set a confirmation message that will be displayed when the button is pressed. Leave it empty if you don't need to display a message.
- CSS Style** CSS for the the button, if you do not change this option Scriptcase will apply the default application/project CSS. You can customize the buttons' CSS using the option within the main menu "Layout » CSS Buttons"
- Reload quantity of records** Option used to update the amount of application records.
- Type** Button type description.
- Target** This option allows you to set the target window where you will run the button code (the same window, other window, modal)

Image

ATTRIBUTE	VALUE
Display Mode	Image
Icon	
Hint	
Confirmation Message	
Reload quantity of records	No
Type	Run
Target	Same Window

Grid's Run Button settings - Image type

- Display Mode** Combo box to select the button display option, you can choose image, button or link.
- Icon** Button display icon. You can use this option to select an image (from Scriptcase images or from you computer) to represents the button within the application toolbar.
- Hint** Using this option you can set a message for the button hint
- Confirmation Message** Using this option you can set a confirmation message that will be displayed when the button is pressed. Leave it empty if you don't need to display a message.
- Reload quantity of records** Option used to update the amount of application records.
- Tipo** Button type description.
- Target** This option allows you to set the target window where you will run the button code (the same window, other window, modal)

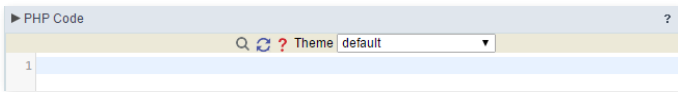
Link

ATTRIBUTE	VALUE
Display Mode	Link
Label	Run
Hint	
Confirmation Message	
CSS Style	default
Reload quantity of records	No
Type	Run
Target	Same Window

Grid's Run Button settings - Link type

- Display Mode** Combo box to select the button display option, you can choose link, button or image.
- Label** Text that will be displayed on the button (on running application).
- Hint** Application button title (text that will be displayed for the button within the application)
- Confirmation Message** Using this option you can set a confirmation message that will be displayed when the button is pressed. Leave it empty if you don't need to display a message.
- CSS Style** CSS class name, style created in the theme buttons editor .
- Reload quantity of records** Option used to update the amount of application records.
- Type** Button type description.
- Target** This option allows you to set the target window where you will run the button code (the same window, other window, modal)

Coding Area



Grid's Run Button settings - Coding area

- here are two types of events in the Run button
 - **nRecord** : Runs after processing on each record selected.
 - **nFinish** : Runs after processing all records selected.

In this coding area you can use Scriptcase Macros, PHP and JavaScript.

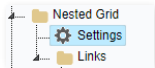
Nested grids are used when you need to display hierarchical data, such as customers and their respective applications. The main Grid encapsulates other Grids, showing in each of its Nested grids additional data records. Note the image below in which for each customer is displayed a sub-query with the applications.

LIST - CUSTOMERS					01/30/2018	
Quick search <input type="text"/>					Export <input type="button" value=""/>	Settings <input type="button" value=""/>
Customerid	Companyname	Contactname	Country			
ALFKI	Alfreds Futterkiste	Maria Anders s	DE			
Orderid	Customerid	Employeeid	Orderdate	Priceorder		
10,643	ALFKI	6	08/25/2008	R\$ 1.321,32		
10,692	ALFKI	4	10/03/2008	R\$ 1.064,80		
10,702	ALFKI	4	10/13/2008	R\$ 399,30		
10,835	ALFKI	1	01/15/2008	R\$ 1.029,71		
10,952	ALFKI	1	03/16/2008	R\$ 595,32		
11,011	ALFKI	3	04/09/2008	R\$ 1.161,60		
11,087	ALFKI	1	05/26/2009	R\$ 47,19		
Grand Summary				R\$ 5.619,24		
ANATR	Ana Trujillo Emparedados y helados	Ana Trujillo	MX			
ANTON	Antonio Moreno Taquerilla	Antonio Moreno	AR			

Customer Grid with an order nested grid.

Settings

In the application menu, by clicking on the item folder Configuration Nested grids, can be changed the General attributes of the Nested grids.



Nested grid menu.

Nested Grid Settings		
ATTRIBUTE	VALUE	DESCRIPTION
Title in the same line	No	Nested Grid title line will be shown on the same line of the application title.
PDF	Yes	Defines if the Nested Grids will appear in the PDF report
XLS	Yes	It defines if the Nested Grids will appear in the XLS report
XML	Yes	It defines if the Nested Grids will appear in the XML report
Enable TreeView	No	Show the Nested Grid in TreeView mode.
Position	In one column	Position of the Nested Grid.

Nested grid general settings.

- tributes**
 - title in the same line** : This attribute enables you to configure the Nested Grid title display. If it is displayed on the same line, Nested Grid table will be incorporated into the main Grid.

Customerid	Companyname	Contactname	Orderid	Employeeid	Orderdate	Priceorder
ALFKI	Alfreds Futterkiste	Maria Anders s	10,643	6	08/25/2008	R\$ 1.321,32
			10,692	4	10/03/2008	R\$ 1.064,80
			10,702	4	10/13/2008	R\$ 399,30
			10,835	1	01/15/2008	R\$ 1.029,71
			10,952	1	03/16/2008	R\$ 595,32
			11,011	3	04/09/2008	R\$ 1.161,60
			11,087	1	05/26/2009	R\$ 47,19
Grand Summary						R\$ 5.619,24
ANATR	Ana Trujillo Emparedados y helados	Ana Trujillo	10,248	10	07/04/2008	R\$ 232,74

Example using Nested Grid title on the same line.

- o **DF**: This option allows you to enable/disable the nested grid display in PDF reports.
- o **enable TreeView** : This option enables an option to hide/display the Nested Grid within the main Grid records. By choosing "Yes", the option "Title in the same line" will be automatically disabled.

Customerid	Companyname	Contactname	
▲ BOLID	Bolido Comidas preparadas	Martin Sommer	
Orderid	Employeeid	Orderdate	Priceorder
10,326	4	10/10/2008	R\$ 1.195,48
10,801	4	12/29/2008	R\$ 4.888,40
10,970	9	03/24/2008	R\$ 338,80
Grand Summary			R\$ 6.422,68

Using tree view in the Nested Grid.

- o **osition** : This option allows you to configure the Nested Grid placement in the main Grid. This option is not available if you enable "Title in the same line". You will be able to position the Nested Grid within the records. The options are:

- **n one column**

Customerid	Companyname	Contactname	Orders
BLONP	Blondesdssl pere et fils	Frederique Citeaux	▶
BOLID	Bolido Comidas preparadas	Martin Sommer	▲
Orderid	Employeeid	Orderdate	Priceorder
10,326	4	10/10/2008	R\$ 1.195,48
10,801	4	12/29/2008	R\$ 4.888,40
10,970	9	03/24/2008	R\$ 338,80
Grand Summary			R\$ 6.422,68
BONAP	Bon appart	Laurence Lebihan	▶
BOTTM	Bottom-Dollar Markets	Elizabeth Lincoln	▶

Displaying the Nested Grid in the same row of the main Grid record.

- **elow the record**

Customerid	Companyname	Contactname	
▲ BOLID	Bolido Comidas preparadas	Martin Sommer	
Orderid	Employeeid	Orderdate	Priceorder
10,326	4	10/10/2008	R\$ 1.195,48
10,801	4	12/29/2008	R\$ 4.888,40
10,970	9	03/24/2008	R\$ 338,80
Grand Summary			R\$ 6.422,68

Displaying the Nested Grid in a separate row of the main Grid record.

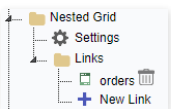
- o **lignment** : Allows you to set the Nested Grid alignment in the main Grid interface when "Position" is set set as Below the record. The options are: left, right and Center.

Set individually

Any of the attributes that use the value "Set individually" requires the property to be configured on each created link settings.

Links

To create a Nested Grid link click on the "New link" within the Link folder.



Creating a new Nested Grid

- o **adding a new Nested Grid** : Enter the name and the label for the link that will be created.

NAME	LABEL
orders	Orders

Buttons: Create, Help

Adding a new Nested Grid

- **Application list** : All Grid applications from current project, that contain global variables, will be listed.

Link between applications - Application List

Select the application that will be called: Search...

Application: All By folder By type

- cons_atend_geral
- cons_nf
- cons_nf_item
- grid_new
- grid_orders
- grid_orders_1
- grid_sales
- orders

Buttons: < Back, Next >, Help

Grid application selection.

- or an application to be used as a Nested Grid, it must have within its SQL command (Grid > SQL) a WHERE clause with a global variable to receive the parameters from the main Grid.

ATTRIBUTE	VALUE
SQL Select Statement	<pre>SELECT orderid, customerid, employeed, orderdate, requireddate, shippeddate, shipvia, freight, priceorder, shipcountry, shipregion, shipstate, shipcity, shipname, shipaddress, shippostalcode FROM orders where customerid = '[customer]'</pre>

SQL query configuration to the Grid be used as Nested Grid.

- **Parameters setting** : Setting the value that will be passed to the variable(s) of the Nested Grid SQL statement.

Link between applications - Parameters Definition

Select values to pass as parameters

PARAMETER	VALUE
customer	Field (customerid)

Options: Field, Fixed, Empty

Buttons: < Back, Save, Help

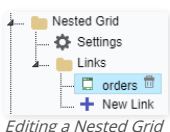
Informing the parameter to the variable

In the screen above, on the left side, it is displayed the Nested Grid input parameters (global variables created within the Nested Grid WHERE clause), on the right side, you must select the parameter option. The options are:

- **tributes**
 - **ield** : Used to assign a Grid field value as a parameter.
 - **ix** : Used to assign a fixed value as a parameter.
 - **mpty** : Choosing this option will not be assigned any value as a parameter.
- **ave button** : By clicking this button the Nested Grid creation will be completed and saved.

Editing an existing Nested Grid link

All Nested Grid links created will be placed within its menu options, inside the folder "Links". By clicking on the Nested Grid name there will be an interface for settings editing.



Editing a Nested Grid

General Settings

General Settings: orders

ATTRIBUTE	VALUE	DESCRIPTION
Label	Orders	Field title.
Link	<div style="border: 1px solid #ccc; padding: 5px;"> <p>Link data</p> <p>Linked to: orders</p> <p>Parameters: customer = {customerid}</p> <p>Edit link Save</p> </div>	Edit Embed Grid Link
Display	<input checked="" type="checkbox"/> Header <input checked="" type="checkbox"/> Sequential <input checked="" type="checkbox"/> Total	Defines which items of the embed application will be shown in the main application.
Visualization	<input checked="" type="checkbox"/>	Defines which embed application items will inherit the visual formatting of the main application

Nested Grid general settings

This interface can be used to edit the following Nested Grid attributes:

- **abel** : Nested Grid title for the application
- **ink** : In this attribute are displayed the current Nested Grid connection data, showing the application and parameters used. To change any link attribute just click to edit Link.
- **isplay** : This attribute allows you to set which Nested Grid items, if it uses any of these resources, will be displayed in the main Grid.

	Orderid	Employeeid	Orderdate	Priceorder
1	10,326	4	10/10/2008	R\$ 1.195,48
2	10,801	4	12/29/2008	R\$ 4.888,40
3	10,970	9	03/24/2008	R\$ 338,80
Grand Summary				R\$ 6.422,68

Display settings

- * **_1_** : Display Title.
- * **_2_** : Sequential display (number line).
- * **_3_** : Display Totals.

- **inherit view** : When marked it sets that the Nested Grid view will be equal to that of the main Grid.

Set Individually

If the value of one of the attributes in the Nested Grid settings has been selected as "Set Individually", these values must be configured in the Nested Grid editing screen for each called link.

General Settings: orders


ATTRIBUTE	VALUE	DESCRIPTION
Label	Orders	Field title.
Link	<div style="border: 1px solid #ccc; padding: 5px;"> <p>Link data</p> <p>Linked to: orders</p> <p>Parameters: customer = {customerid}</p> <p>Edit link Save</p> </div>	Edit Embed Grid Link
Display	<input checked="" type="checkbox"/> Header <input checked="" type="checkbox"/> Sequential <input checked="" type="checkbox"/> Total <input type="checkbox"/> Title in the same line	Defines which items of the embed application will be shown in the main application.
Position	In one column	Position of the Nested Grid.
Enable TreeView	<input type="checkbox"/>	Show the Nested Grid in TreeView mode.
PDF	<input type="checkbox"/>	Defines if the Nested Grids will appear in the PDF report
Visualization	<input type="checkbox"/>	Defines which embed application items will inherit the visual formatting of the main application

Nested Grid settings.

Display Settings

These options will only appear if all of the following conditions are true:

Attribute	Value
Title in the same line	No
Enable Tree view	No
Position	In one column

Display Settings		
Horizontal Alignment	<input type="text"/>	Field horizontal alignment.
Vertical Alignment	<input type="text"/>	Field vertical alignment.
Background Color	<input type="text"/> 	Background Color
Title Horizontal Alignment	<input type="text"/>	Title horizontal alignment.
Title Vertical Alignment	<input type="text"/>	Title vertical alignment.
Bold	<input type="checkbox"/>	Display the field title in bold.

Nested Grid display settings

- **tributes**
 - **orizontal Alignment** : This option sets the Nested Grid horizontal alignment. It can be aligned left, right or Center.
 - **ertical Alignment** : This option sets the Nested Grid vertical alignment. It can be aligned to the top, middle, or End.
 - **ackground color** : Here you can set the background color. If it is not filled it will get the Grid default background color.
 - **itle horizontal alignment** : This option sets the Nested Grid label horizontal alignment. It can be aligned Left, Right or Center.
 - **itle vertical alignment** : This option sets the Nested Grid label vertical alignment. It can be aligned to the top, middle, or End.
 - **old** : This option sets the Nested Grid title (label) in bold.

The total fields will be displayed only when the Grid application is using at least one Group By.

Settings

The settings below only can be applied for the Grid totals.

Summarization Functions		
ATTRIBUTE	VALUE	DESCRIPTION
Results in a single line.	<input checked="" type="checkbox"/>	Display the Main Total title and its value on a single line.
Display Total	On every page ▾	Pages where the total will be displayed.
Group Subtotal	Below ▾	Display the Group subtotal after the records of the Group By.
Record Count	<input checked="" type="checkbox"/>	Display the record count in the total.

- **results in a single line.**: This option sets the display of the **General Total** title and its results in a single line.

- example for this option **Enabled**:

Grand Summary(872) - Sum	R\$ 267.620,19
- Max	R\$ 4.791,60
- Min	R\$ 0,00

- example for this option **Disabled**:

Grand Summary (872)	
Sum	R\$ 267.620,19
Max	R\$ 4.791,60
Min	R\$ 0,00

- **isplay Total** : This option indicates in which pages the General Total will be displayed. The options are: **On every page**, **On the last page** or **Do not display**.
- **roup Subtotal** : This option defines where the subtotal group by will be displayed.
- **ecord Count** : This option allows you to view the amount of records by the General Total title.
 - example for this option **Enabled**: **Grand Summary(872)** * Example for this option **Disabled**: **Grand Summary**

Layout settings

Setting of display of the subtotal group.

Layout settings		
reg1		
GROUP BY FIELD	LABEL	DISPLAY
date_YYYYMMDD2	Group Subtotal	<input checked="" type="checkbox"/>
reg2		
GROUP BY FIELD	LABEL	DISPLAY
region		<input checked="" type="checkbox"/>

- **abel** - This field sets the Label for the **Group Subtotal**.
- **isplay** - This option sets if the **Group Subtotal** it will be display.

Select fields

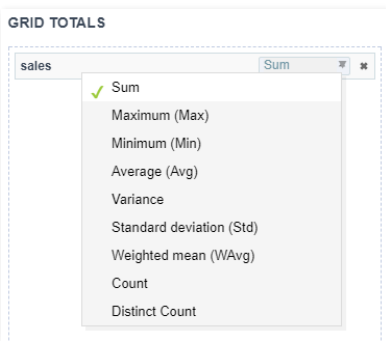
Using the select fields area you can set the total fields and the total options for each field. The same field can be used more than once in the Grid totals area, just if they are using different summaries types.

To define the fields that will be used for totals, drag them to the area, **Grid Totals**.



Note: The total fields will only be displayed if they are also selected to be displayed within the Grid application module.

When positioning the fields, you must define what summarization will be used, to select that you must click in the combo box and select one of the available options, that will be according to the data type (integer, date, text...).



The summarization options available are:

- **um** : Sets a sum of the values for the selected field.
- **aximum** : Displays the highest value identified in the selected field.
- **inimum** : Displays the lowest value identified in the selected field.
- **verage (Avg)** : Calculates the arithmetic mean of the values for the selected field.
- **ariance** : Calculates the dispersion of the values related to the average.
- **tandard Deviation** : Measures the variability of values around the average, the minimum value of the standard deviation is 0 indicating that there is variability, i.e. that all values are equal to the mean.
- **ighted mean (WAvg)** : Calculates the weighted average for the selected field. To set the weight used in the calculation of the average access field settings selected in **Totals > Fields (select the field where you are using the Weighted mean) > Weighted average weight**.
 - **ighted average weight** : Field that will be used as weighted average weight. In calculating the weighted average, each set value is multiplied by its "weight", that is its relative importance.
- **ount** : Displays the total number of records for the selected field.
- **istinct Count** : Displays the total number of records for the selected field, distinguishing the values.

Important note: The fields in the Grid totals are displayed only on Grids with no Group by or with empty Group by.

Positioning

Defines the positioning and the label used by Grid totals. There are three display formats,, **Default**, **Grouped** or **By field**

Default

To add the same field two or more times in the Totals or use different types of summaries in selected fields, this option will not be displayed.

The option **Default** returns the result below the column being summarized. When you use the total for more than one column of the Grid, using the same type of summarization, the results are displayed on the same line.

Summary Fields Order

Default
 Grouped
 By field
 Alignment: Left

Illustration

FIELD 1	FIELD 2	FIELD 3	FIELD 4	FIELD 5
001	Testing...	Testing...	\$ 7,552.96	\$ 1,759.83
002	Testing...	Testing...	\$ 97.16	\$ 22.63
003	Testing...	Testing...	\$ 5,568.00	\$ 1,297.34
Total			\$ 13,218.12	\$ 3,079.82

The line with **General Total** displaying the **Sales**.

Id	Date	Region	Product Category	Sales
1	01/01/2015	South Atlantic	SEAFOOD	R\$ 50,82
2	01/01/2016	South Atlantic	CONFECTIONS	R\$ 84,70
3	01/01/2013	South Atlantic	BEVERAGES	R\$ 90,75
4	01/02/2013	East South Central	CONFECTIONS	R\$ 163,35
5	01/02/2014	East South Central	DAIRY PRODUCTS	R\$ 251,68
Grand Summary				R\$ 267.620,19

Alignment

This option is available for display formats **Default** and **Grouped**.

Defines the placement of the label selected within the **Setting > Display total** option. The alignment can be in the **Center**, **Left** e **Right**:

- eft** : Grand Summary R\$ 267.620,19
- ight** : Grand Summary R\$ 267.620,19
- enter** : Grand Summary R\$ 267.620,19

Label settings (Default)

This option allow you to customize the total labels.

As default we are going to display Grant total

Label settings (Default)

ATTRIBUTE	VALUE	DESCRIPTION
Total line message	<input type="text" value="(lang_msgs_tot)"/>	Message displayed in the line of total

Grouped

The option **Grouped** returns the total result, separating each type per line.

It is possible to position the total lines by dragging to the desired position.

Summary Fields Order

Default
 Grouped
 By field
 Alignment: Left

FIELD 1	FIELD 2	FIELD 3	FIELD 4	FIELD 5
001	Testing...	Testing...	\$ 550.00	\$ 200.00
002	Testing...	Testing...	\$ 150.00	\$ 350.00
+ Sum			\$ 700.00	\$ 550.00
+ Average (Avg)			\$ 350.00	\$ 275.00
+ Maximum (Max)			\$ 550.00	\$ 350.00
+ Minimum (Min)			\$ 150.00	\$ 200.00
+ Count			\$ 250.00	\$ 470.00
+ Distinct Count			\$ 95.00	\$ 300.00
+ Variance			\$ 50.00	\$ 120.00
+ Standard deviation (Std)			\$ 200.00	\$ 900.00
+ Weighted mean (WAVg)			\$ 150.00	\$ 500.00

In this example, we are displaying the sum, average, maximum and minimum of column **Sales**.

Id	Date	Region	Product Category	Sales
1	01/01/2015	South Atlantic	SEAFOOD	R\$ 50,82
2	01/01/2016	South Atlantic	CONFECTIONS	R\$ 84,70
3	01/01/2013	South Atlantic	BEVERAGES	R\$ 90,75
4	01/02/2013	East South Central	CONFECTIONS	R\$ 163,35
5	01/02/2014	East South Central	DAIRY PRODUCTS	R\$ 251,68
Grand Summary				
Sum				R\$ 267.620,19
Avg				R\$ 306,90
Max				R\$ 4.791,60
Min				R\$ 0,00

Alignment

This option is available for display formats **Default** and **Grouped**.

Defines the placement of the label selected within the **Setting > Display total** option. The alignment can be in the **Center, Left e Right**:

- **eft** : **Grand Summary** R\$ 267.620,19
- **ight** : **Grand Summary** R\$ 267.620,19
- **enter** : **Grand Summary** R\$ 267.620,19

Label settings (Grouped)

This option allow you to customize the total labels.

ATTRIBUTE	VALUE	DESCRIPTION
Sum Label	<input type="text" value="{lang_btms_smy_msge_sum}"/>	Sum function title
Average Label	<input type="text" value="{lang_btms_smy_msge_avg}"/>	Average function title
Max Label	<input type="text" value="{lang_btms_smy_msge_max}"/>	Function max title
Minimum Label	<input type="text" value="{lang_btms_smy_msge_min}"/>	Min function title.
Count Label	<input type="text" value="{lang_btms_smy_msge_cnt}"/>	Title function Count
Distinct Count label	<input type="text" value="{lang_btms_smy_msge_dct}"/>	Title function Distinct Count
Variance Label	<input type="text" value="{lang_btms_smy_msge_var}"/>	Title function Variance
Standard Deviation Label	<input type="text" value="{lang_btms_smy_msge_pad}"/>	Title function Standard Deviation
Weighted mean label	<input type="text" value="{lang_btms_smy_msge_wei}"/>	Label for the weighted mean title

By Field

This option displays the values in the left corner by positioning the results next to each other.

Summary Fields Order				
<input type="radio"/> Grouped <input checked="" type="radio"/> By field <input type="checkbox"/> Break line per field				
Illustration				
FIELD A	FIELD B	FIELD C	FIELD D	FIELD E
001	Testing...	Testing...	\$ 400.00	\$ 100.00
002	Testing...	Testing...	\$ 250.00	\$ 350.00
Total				

Line break per field

This option sets the total fields display. If selected it shows the fields in the same row or divided by line.

Separated by line:

Grand Summary

Sales (Sum) R\$ 267.620,19
Profit (Sum) (\$803,977.68)

Displayed on the same line:

Grand Summary

Sales (Sum) = R\$ 267.620,19 Profit (Sum) = (\$803,977.68)

Label settings (By field)

This option allow you to customize the total labels.

Label settings (By field) ?			
ATTRIBUTE	ATTRIBUTE	VALUE	DESCRIPTION
Totals	Label (sales sum)	<input type="text" value="Sales ((lang_btms_smyr_msge_sum))"/>	Título do campo.
	Label (profit sum)	<input type="text" value="Profit ((lang_btms_smyr_msge_sum))"/>	Título do campo.

Fields

This option allows you to configure the display of values and labels of the selected fields in the Grid totals.

General Settings

Allows you to change the label displayed in the totals

General Settings: sales_sum		
ATTRIBUTE	VALUE	DESCRIPTION
Label	<input type="text" value="Sales ((lang_btms_smyr_msge_sum))"/>	Título do campo.
Sales (Sum)		

Visual Configuration For The Grid Totals

Sets the formatting of the fields displayed in the subtotal for the group.

Visual settings of totalization in a Group By		
ATTRIBUTE	VALUE	DESCRIPTION
Text font	<input type="text" value="Aa"/>	Text field font
Font Size	<input type="text" value="v"/>	Font Size
Font Color	<input type="text" value="c"/>	Font Color
Background Color	<input type="text" value="c"/>	Background Color
Text bold.	<input type="text" value="v"/>	Formatting text bold.

- **ont family** : Sets the font used.
- **ont size** : Sets the font size.
- **ext color** : Sets the text color
- **ackground color** : Sets the background color.
- **old** : Format text in bold.

Example of formatting the subtotal for the Group:

Date of Date => 01/09/2013					
Id	Date	Region	Product Category	Sales	Profit
22	01/09/2013	East South Central	BEVERAGES	R\$ 163,35	\$36.30
19	01/09/2013	East South Central	GRAINS/CEREALS	R\$ 363,00	\$121.00
25	01/09/2013	East North Central	CONFECTIONS	R\$ 116,16	\$19.36
Total sales on the day - 01/09/2013				R\$ 642,51	

General Total Visual Setting

Sets the formatting of the fields displayed in the General total.

Visual setting of the General Total		
ATTRIBUTE	VALUE	DESCRIPTION
Text Font	<input type="text" value="Aa"/>	Field Text size
Font Size	<input type="text" value=""/>	Font Size
Font Color.	<input type="text" value=""/>	Font Color.
Background Color	<input type="text" value=""/>	Background Color
Bold text	<input type="text" value=""/>	Formating text in bold.

- **ont family** : Sets the font used.
- **ont size** : Sets the font size.
- **ext color** : Sets the text color
- **ackground color** : Sets the background color.
- **old** : Format text in bold.

Example of formatting the Grid General Total

Id	Date	Region	Product Category	Sales	Profit
1	01/01/2015	South Atlantic	SEAFOOD	R\$ 50,82	\$14.52
2	01/01/2016	South Atlantic	CONFECTIONS	R\$ 84,70	\$33.88
3	01/01/2013	South Atlantic	BEVERAGES	R\$ 90,75	\$36.30
4	01/02/2013	East South Central	CONFECTIONS	R\$ 163,35	\$54.45
5	01/02/2014	East South Central	DAIRY PRODUCTS	R\$ 251,68	\$94.38
Grand Summary				R\$ 267.620,19	

Settings

Navigation

This interface allows defining the navigating behavior of the application

ATTRIBUTE	VALUE
Exit URL	<input type="text"/> 🔗
Close on Exit	<input type="checkbox"/>
Redirect URL	<input type="text"/> 🔗
Redirect Variable	<input type="text"/>

Navigation Interface.

Exit URL

URL to where the user goes when he clicks on the “exit” button.

Close on Exit

Close the browser window when the user clicks on the “exit” button.

Redirect URL

Redirect to another URL in case there aren't any global variables available.

Redirect Variable

Creates a variable with the application name and sends it to the redirected application.

Global Variables

This screen shows all global variables used in the application

The global variable is an external parameter required for running the application. The application can include global variables in the WHERE clause, field definitions and names, event programming, and so on.

You can define global variables by using square brackets ([variable]). You can pass the parameters to the application through one of the methods: Session, Post, and Get

if you need to use [Database Identifiers](#) in or tables, we recommend using the double quotation marks(" ") instead of the brackets([]) because of the conflict with the [Global Variables syntax](#) of Scriptcase. Using the brackets as database identifiers may lead to problems in the Scriptcase applications.

APPLICATION
<ul style="list-style-type: none"> ⚙ Settings 🔗 Navigation 📄 Messages 🌐 Global Variable

Global variables Interface.

Example:

```
Select CustomerID, CustomerName from Customers WHERE CustomerID = '[v_customerid]'
```

We do not recommend using session variables (global variables) with the same name as the table fields.

In this case, the variable **v_customerid** is displaying in the global variables configuration.

ATTRIBUTE	VALUE
global	<div>Scope</div> <input type="checkbox"/> SESSION <input checked="" type="checkbox"/> POST <input checked="" type="checkbox"/> GET
	<div>Settings</div> <input type="checkbox"/> Optional
	<div>Type</div> <input type="radio"/> Out <input checked="" type="radio"/> In

Global variables configuration Interface.

Attribute

It shows the variable's name in the application.

Value

Defines the behavior of the variables, divided into three blocks, they are:

Scope

Defines how the application receives the variable. A variable defined as the POST method in the app does not receive value if it comes with the GET method. Those are the methods available:

- **ession**: Defines that the variable must be created as a PHP session variable by another application.
- **ET**: Defines that the variable must be passed by the URL, that is, visible by the browser.
- **OST**: Defines that the variable must be passed through the POST method of an HTML form.

Settings

- **ptional**: Check it to don't validate the variable at runtime.

Type (In / Out)

Defines whether the variable comes from another application (In), or it is starting in the current app (Out).

Description

Indicates where the application is using the variable.



1. [Home](#)
2. Applications

General Overview

Allows to create link between applications that are in the same project, the link option incorporates various functionalities like above.

New Link

To create a new link you just need to access the option **Link between Applications** located in the menu at the left side.

When clicking in **New Link** the screen below will be display:

When clicking in **New Link** the screen to choose the type of link that will be created will be displayed.

[Edit Link:](#)

Allows to edit registers of the grid using a form.

[Capture Link:](#)

Allows to create a link specifically to the filter's fields, capturing data from other grid.

[Field Link:](#)

Allows to create a link represented through a link, from a grid's column to any application from the project.

[Button Link:](#)

Allows to create a link through grids buttons to any application from the project.



1. [Home](#)
2. [Applications](#)
3. Stored Procedures
4. New link

New link

![[Links menu interface.]](ligacao_nova) *Links menu interface.*

When you click the "New Link" item from ScriptCase application menu will, the screen below will be displayed.

![[Interface for creating links]](ligacao_criando) *Interface for creating links*

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. Application Link

Application Link

Used to edit records of a Grid Application by using a Form Application. In the generated Grid you will view a link for each record to edit the records. Clicking on the link, the form selected can be displayed in various ways (in an iframe, in the same window or on another window).

List of Applications

Application: You need to select the form application that is going to be called by the Grid Application.

Parameters Definition

Parameters Definition Interface.

On the screen above, on the left side, are displayed the parameters that are expected for the form application (Primary Key, Global Variables), on the right side, you need to select the option that is going to be set to the parameter. The options are:

- **Field** : Used to pass a field value of a Grid as a parameter.
- **Value** : Used to pass a static value as a parameter.
- **Variable** : Used to pass a global variable value used in the Grid as a parameter.
- **Empty** : Choosing this option, no value will be passed as a parameter.



1. [Home](#)
2. [Applications](#)
3. Stored Procedures
4. Fiels Link

Fiels Link



1. [Home](#)
2. [Applications](#)
3. Stored Procedures
4. Capture Link

Capture Link



1. [Home](#)
2. [Applications](#)
3. Stored Procedures
4. Button Link

Button Link



1. [Home](#)
2. [Applications](#)
3. Stored Procedures
4. Links Edit

Links Edit

In the Link Folder of the Application Menu (Image Below) are displayed the links existing in the application and also the item New Link. When clicking on the existing link it is displayed the screen below that allows to manage the links.

 *Editing Links.*

Actions

Properties Change the link's behavior, position, and how the link opens.

Link Change the application that's being called in the link their parameters.

Delete Remove the existing link.

- 1. [Home](#)
- 2. [Applications](#)
- 3. Stored Procedures
- 4. Link Properties

Link Properties

On the screen below, you can see the attributes relevant to the link behavior of that need to be informed.

Link properties Interface.

- **Link Operation Mode :**
 - **Open in the same Window :** Displays the Form in the same browser window.
 - **Open in another Window :** Displays the Form in another window of the browser.
 - **Open in IFRAME :** The Form will be displayed in the same window, allowing to position below, above, to the left or right of the Grid.
 - **Open in Parent :** If the Grid is in a iframe, it displays the Form in the page's parent view port.
 - **Modal :** Opens the Form in modal, allowing to configure the modal size.
- **Display button (new row) on the grid toolbar :** This option, when enabled, adds a button to the toolbar that allows to add a New Record to the Form.
- **Label for the button New :** You can apply a Label for the button, if not informed the label will be "New".
- **Hint for the button New :** You can apply a Hint for the button, if not informed the hint will be "New".
- **Hotkey for the button "New" :** Indicates which key will be the shortcut for the New Record.
- **Exit URL for the target application :** The URL that is going to call after exiting the Form. In case no value is informed, the "back" button will redirect to the Grid.
- **Form Property**

Allows to select the buttons that are going to be displayed in the Form through the link. *Form properties configurations Interface.*

- **Enable insert button on target application :** Enables the Insert Button in the Form.
- **Enable update button on target application :** Enables the Update Button in the Form.
- **Enable delete button on target application :** Enables the Delete Button in the Form.
- **Enable navigation button on target application :** Enables the navigation buttons (first, previous, next and last) in the Form.
- **Enable button to edit a grid record :** Enables the edit button for each record.
- **IFrame properties**

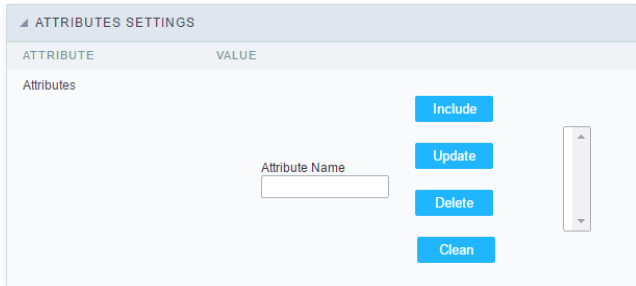
These options are displayed when the property Link Operation Mode is set to "Open in IFRAME". *IFrame properties configurations Interface.*

- **Display the header of the called application :** Displays the Form's header.
- **IFrame position relative to the main application :** It can be: below, above, right or left.
- **Action After Insert view the list below :**
 - **Reload Grid :** Does a refresh on the current page.
 - **Move to the end of the grid :** Navigate to the Grid's last page.
- **IFrame height :** IFRAME Height in pixels.
- **IFrame width :** IFRAME Width in pixels.

The concept of programming is incorporated in this ScriptCase version. With the use of resources of attributes, methods and libraries. In the previous version was already possible create business rules in the applications, using this concept. The biggest difference is that now this can be realized in a more organized way, making it easier to be developed and comprehended by another developer.

Attributes

The attributes are variables that has the global scope declared in the application. An attribute that can be referenced in all the methods and the application's events.



Attributes configuration Interface.

- **attributes**
 - **attribute Name** : This field allows to inform the attributes name, after typing the name just click on **include**. In case you want to **update** click on the attributes name (located on the buttons' right side), do the necessary modifications and click on update. In case it is necessary to **delete** an attribute, it is necessary to do same process as the update, but when selecting the attribute, it is necessary to click on the delete button. The **clear** button, as the name suggests, it clears the list of all the attributes listed on the right.

Internal Libraries

To use the internal libraries just select the correspondent (Project Libraries for example). This process allows the methods existing in the library, to be visible in all the application's Events and Methods. See how to manage the libraries by [clicking here](#).

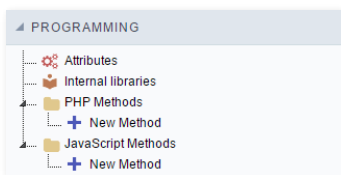


Internal Libraries management Interface

PHP Methods

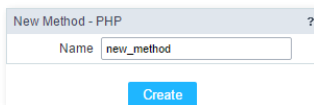
Methods are function or procedures declared by the developer, that helps when applying the routines. Using methods in the applications allows to reuse your code throughout the application, optimizing the development experience.

Creating A New Method



- Give the method a name and click Create. As in the image below.

PHP methods are incorporated into the same class, so it is not possible to use two functions with the same name even when used in different methods.



- Methods can receive parameters.

```
function new_method
1 echo "Hello World!!!";
```

- dd the amount of variables:

Definition of the parameters of the method:

new_method

- No defined parameter.

Add Parameter(s) Cancel

Save

- defining the variables:

Insertion of Parameters

Name	Type	Value Standard
<input type="text"/>	For Value	<input type="text"/>
	For Value	
	For References	

Save Back Cancel

- **ame** : Type in the variable's name.
- **ype** : Selecting the type of variables: For Value or For Reference.
- **alue Standard** : The parameter's value used to initialize when calling the method.

References allows to create a second name for a variable that you can use to read and modify the variable's original information.

- diting a parameter:

Definition of the parameters of the method:

new_method

Parameters

Stest = test

⏪ ⏩

☑ ☐ ✎ ✖

Add Parameter(s) Cancel

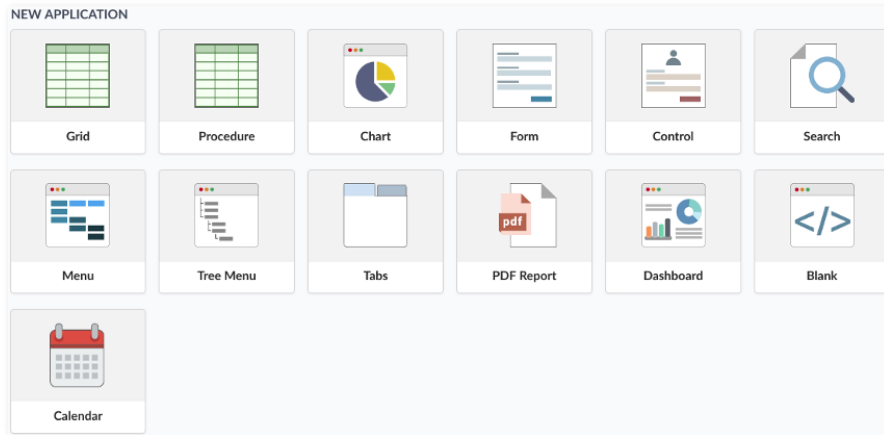
Save

- ☑ : Checking all the parameters in the parameter list.
- ☐ : Uncheck all the parameters in the parameter list.
- ✎ : Edit the selected parameter of the list.
- ✖ : Deletes the selected variable of the list.

New Application

The creation process of one application is simple enough and intuitive.

You just need to select which application you want to generate in the applications available in Scriptcase, That it will be generated automatically.



Dados da Aplicação

When the Procedure application is selected the following screen will be displayed:

In this tab you need to select the connection that will be used in the application. When selected the connection, you need to inform the name of the procedure that will be used and inform if it uses parameters.

The image shows the 'PROCEDURE' configuration screen. It has two tabs: 'APPLICATION DATA' (selected) and 'THEME'. The 'APPLICATION DATA' tab contains the following fields:

- Procedure name:
- Connection: (dropdown)
- Name: (required, indicated by a red asterisk)
- Localization: (dropdown, required, indicated by a red asterisk)
- Number of parameters:

- **Procedure Name:** You will inform here the procedure name that will be used in this application.
- **Connection:** Select the connection SQL Server that has the called procedure.
- **Name:** It defines the name of your Procedure application.
- **Localization:** It defines the language and charset that the application will use.
- **Number of parameters:** Informs how many parameters to be defined exists in the Procedure.

Edit Fields

In this tab we will set the parameters informed in the application creation, to be defined correctly.

- **Parameter:** It shows the quantity of parameters informed in the application's data.
- **Var Name:** It informs the variable name to be passed as a parameter in the Procedure.
- **Type:** Selects the type of data of the parameter.
- **Value:** It defines a default value to be called by the parameter.

Theme

In this screen you select the theme that the application will use. The default project's theme, that can be checked and changed in **Project > Properties**, will be selected automatically.

- **Theme** - List of available themes in the project.

*After all the settings are done, click in the button **Create**. In the next screen the settings of the created application will be displayed.*

It allows you to send the generated export file by email. For that, you may be using SMTP or integration with **Mandrill** and **SES**.

For more information about Mandrill [click here](#)

For more information about Amazon SES [click here](#)



Sending Options

We must configure how to send the email or select an API already set in Tools> API. [Click Here](#) and see how to configure.

SMTP

SMTP is the standard protocol for sending emails over the Internet, and each provider has its SMTP.

For more information about SMTP settings [click here](#)

Sending settings		
ATTRIBUTE	VALUE	DESCRIPTION
API	- Custom -  	API for sending e-mails.
Gateway	smtp	
SMTP server	smtp.example.com	SMTP server to send email.
SMTP port	465	SMTP server port (example: 25, 465, 587).
SMTP user	root	User to connect to the SMTP
SMTP password	*****	SMTP server connection password.
SMTP protocol		Encryption protocol used by the SMTP server.
From email	default@example.com	Default shipping email.
From name	default	Default shipping name.

- **PI:** Allows you to select an API that is already configured in **Tools> API** or set a new one here by selecting **- custom -**. In this case you will see some options according to the selected Gateway.

SMTP Server

SMTP server address for the used provider.

Port SMTP

Define the Port of the mail server. Use the port 465 for security with SSL, 587 for security with TLS, or port 25 for port without security. By default, Scriptcase uses port 25.

User SMTP

SMTP server user.

Enter SMTP

SMTP server user password.



Protocol SMTP

Select the security protocol. If no value is declared, Scriptcase uses the No Security protocol as default.

- **-mail:** Sets the origin email, which will send the emails.
- **ame:** Name that will be displayed in the sent email.

Mandrill

Mandrill is a transactional email API for MailChimp users, ideal for sending data-driven emails.

Sending settings		
ATTRIBUTE	VALUE	DESCRIPTION
API	- Custom -  	API for sending e-mails.
Gateway	mandrill	
API key	Your API	API connection key.
From email	default@example.com	Default shipping email.
From name	default	Default shipping name.

- **PI:** Allows you to select an API that is already configured in **Tools> API** or set a new one here by selecting **- custom -**. In this case you will see some options according to the selected Gateway.
- **PI KEY:** Enter the key you obtained when setting up your API.
- **-mail:** Sets the origin email, which will send the emails.
- **ame:** Name that will be displayed in the sent email.

Amazon SES

Amazon Simple Email Service ([Amazon SES](#)) is an email sending service designed to assist in sending marketing emails, notifications, and transactional messages.

Sending settings		
ATTRIBUTE	VALUE	DESCRIPTION
API	- Custom - <input type="button" value="🔄"/> <input type="button" value="✎"/>	API for sending e-mails.
Gateway	Amazon SES	
API key	Your Key API	API connection key.
API Secret	Your Secret API	API secret, obtained in association with API KEY.
Region	Region	API Connection Region.
From email	default@example.com	Default shipping email.
From name	default	Default shipping name.

- **PI**: Allows you to select an API that is already configured in **Tools> API** or set a new one here by selecting **- custom -**. In this case you will see some options according to the selected Gateway.
- **PI KEY**: Enter the key you obtained when setting up your API.

API Secret

Enter the secret access key of your Amazon SES account.

Region

Amazon SES has endpoints in several regions, to reduce network latency, inform the region of the endpoint closest to your application. [See the regions.](#)

- **-mail**: Sets the origin email, which will send the emails.
- **ame**: Name that will be displayed in the sent email.

Export Settings

In these settings, we define the settings of the email sending interface when exporting, and we can define a default email subject and body.

Export settings		
ATTRIBUTE	VALUE	DESCRIPTION
To	<input type="text" value="[mail]"/>	Enter the default value for the field 'To'.
It has copy	Yes	Sets whether to display the field 'copy' within exporting options.
Copy (cc)	<input type="text" value="[mailcopy]"/>	Enter a default value for the field 'Copy'.
It has blind carbon copy (bcc)	Yes	Sets whether to display the blind carbon copy field in export option.
Blind carbon copy	<input type="text" value="[mailbcc]"/>	Enter the default value for the field 'blind carbon copy'.
Title	<input type="text" value="{lang_export_email_subject} sales"/>	Default email subject text. You can change the lang variable or add fixed text.
Search of sales		
Body	<input type="text" value="{lang_export_email_body}"/>	Default text for the body of the email. You can change the lang variable or add a fixed text.
Hello, Please find attached with this email the report in %. Thank you and Have a Nice Day		

To

Enter the default destination email.

Use Copy (CC)

Defines whether or not to display the 'copy' field within the export options.

- **o**: There will be no possibility to send email with copies.
- **es**: The field will be displayed within the export options and we can define a default email for copying. The end user can view this email.
- **idden**: The field will not be displayed within export options and we can set a default email for copying.

Copy (CC)

Enter the default email for the copy field. This option can also be empty, or you can use a global variable containing an email list.

Using Hidden Copy (BCC)

Sets whether or not to display the 'hidden copy' field within the export options.

- **o**: There will be no possibility to send email with copies.
- **es**: The field will be displayed within the export options and we can define a default email for copying. The end user can view this email.
- **idden**: The field will not be displayed within export options and we can set a default email for copying.

Copy (BCC)

Enter the default email for the hidden copy field. This option can also be empty, or you can use a global variable containing an email list.

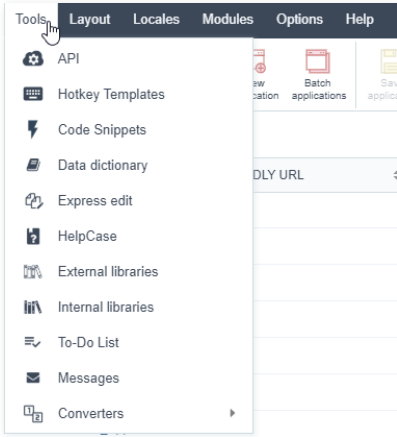
Subject

Default email subject text. You can change the lang variable or add static text.

Color

Default body text for the email. You can change the lang variable or add static text.

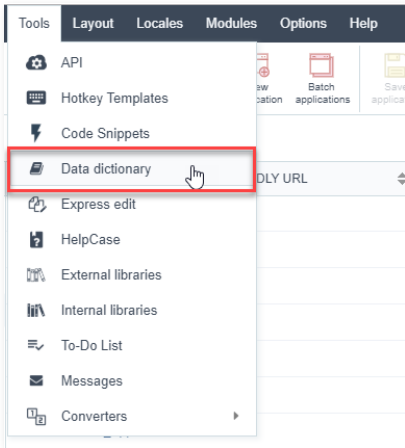
Scriptcase offers various tools that help you out on developing systems to optimize the time of development, making it easier to create and maintain your applications within Scriptcase. In this tutorial, you learn what the environment tools are.



- **Data Dictionary** - You can use it to standardize the tables fields. It helps you to manage the language variables so you can translate the applications created within Scriptcase as well as model the fields to create the applications faster.
- **Express Edit** - This option helps you to edit various applications simultaneously, it contains most of the application options for edit.
- **External Libraries** - This tool offers the option to include external libraries to your project like JavaScript, CSS, HTML, and other libraries. It allows you to integrate the various libraries to your applications.
- **Internal Libraries** - This tool offers the option to create PHP functions to be used in the applications.
- **To-Do List** - Allows you to create a To-Do List and assign Tasks to your developers.
- **Messages** - Allows the developers to exchange messages between themselves in the same ScriptCase.
- **Converters** - This tool offers the option to convert your projects from previous versions of ScriptCase to the current version.
- **Database Import** - This tool offers the option to import tables in the formats (XLS, CSV, and ACCESS) for the following databases: MySQL, PostgreSQL, SQLite, and SQL Server
- **SQL Builder** - This tool is an interface that provides some options to create a **SELECT** query to be used in your applications.
- **Database Builder** - This tool is an interface to manage your database directly from ScriptCase, without needing to use any other software. It supports the Databases: MySQL, SqlServer (not ODBC), SQLite, Oracle, PostgreSQL.

In this article, you learn how to use the Data Dictionary. It is a tool used to assist you with the application translation as well as to standardize the database fields. With the data dictionary, you can create a data repository and associate it to one or more tables.

You can access the data dictionary from the main menu **Tools > Data Dictionary**.



Creating a Data Dictionary

Inform the necessary data for the creation of the repository. The files with an asterisk are required.

DATA DICTIONARY ?

Data Dictionary Name *

Description

Use schema before the table

Convert tables/fields to

Connection *

Associate dictionary with connection *

Main features:

- Analyze your table structures (reverse engineering) and store it in a Dictionary.
- Allows to change the name, size and type of table fields stored in the dictionary.

*The option **Associate dictionary with connection**, define a connection to associate with the repository data, allowing that the new applications, created using the tables associated with the repository, have indexes in the language files with the field labels from the tables.*

Add Tables To The Data Repository

Here we need to select the tables to add to the repository. After selecting the desired tables, click on next.

ADD TABLES TO DICTIONARY

Select the database tables to add to dictionary and click Next to continue.

Table	LABEL
<input checked="" type="checkbox"/> account	<input type="text" value="account"/>
<input checked="" type="checkbox"/> application_logs	<input type="text" value="application_logs"/>
<input checked="" type="checkbox"/> bar_code	<input type="text" value="bar_code"/>
<input checked="" type="checkbox"/> categories	<input type="text" value="categories"/>
<input checked="" type="checkbox"/> city	<input type="text" value="city"/>

You should choose whether or not to create the indexes (langs) on the field labels of the existing applications of the project.

ADD TABLES TO DICTIONARY

If your applications use multiple languages, Scriptcase can automatically create indexes for the application fields labels in the language files.

Generate indexes in language files with table field labels?

Overwrite language files indexes ?

- generate the indexes in the language files of the table fields - On this option, you replace the field labels with the indexes that can be edited in one place when accessing **Locales > Application Language**.

An example of an index created by the Data Dictionary:

Label	{lang_login_user}
	User

- overwriting the indexes in the language files - On this option, you replace the existing indexes with new data based on the database.

Next, we can view a list of tables added to the repository data, where we can view a couple of information about these tables, as the name of the table of the database and the index created for each table, on the label field, and the version of the Data Dictionary of each field.

Dictionary tables Database tables

TABLE	LABEL	VERSION	
<input type="checkbox"/> ● account	{lang_tbi_account}	1	Edit
<input type="checkbox"/> ● application_logs	{lang_tbi_application_logs}	1	Edit
<input type="checkbox"/> ● bar_code	{lang_tbi_bar_code}	1	Edit
<input type="checkbox"/> ● categories	{lang_tbi_categories}	1	Edit
<input type="checkbox"/> ●	Edit

The tables that not added can be viewed when accessing the **Dictionary tables**; the process of including these tables to the repository of data is the same.

When doing a modification to the structure of the database tables, the same should be updated using the option **SYNCHRONIZE DICTIONARY**.

Table Status

Next to the name of the tables, there's an icon that indicates the status of the tables.

- green - Indicates that the tables and the repository are updated.
- red - Indicates that the tables don't exist in the database. In this case, you should delete it from the repository.
- yellow - Indicates that the tables are not updated because of some modification in its structure. In these cases, you should click on the link **edit** referring to the table and click on update, so that the data of the repository could update.

Edit Tables

You can find it on the right side, in the list of tables.

TABLE	LABEL	VERSION	
<input type="checkbox"/> ● bar_code	{lang_tbi_bar_code}	1	Edit

This option allows you to edit the fields from the chosen table, like the data types and the indexes created by the repository, for example.

FIELD	DATA TYPE	LABEL	LENGTH (DB)	DECIMALS
<input checked="" type="checkbox"/> id	Number	{lang_bar_code_fid_id}	0	0
<input checked="" type="checkbox"/> codabar	Text	{lang_bar_code_fid_codabar}	128	0
<input checked="" type="checkbox"/> code11	Text	{lang_bar_code_fid_code11}	128	0
<input checked="" type="checkbox"/> code39	Text	{lang_bar_code_fid_code39}	128	0
<input checked="" type="checkbox"/> code39_ext	Text	{lang_bar_code_fid_code39_ext}	128	0
<input checked="" type="checkbox"/> code93	Text	{lang_bar_code_fid_code93}	128	0
<input checked="" type="checkbox"/> code128	Text	{lang_bar_code_fid_code128}	128	0

Table: bar_code Version: 1 Update Back

Other properties of the fields can be accessed when clicking the plus icon on the right side of the field list.

Required	Minimum size	Maximum size	Min value	Max value
No	0	0	0	0
Mask	Help			

Syncing Dictionary

This option is activated when selecting one or more tables; on this option, you update the data dictionary according to the database.

ADD TABLES TO DICTIONARY

If your applications use multiple languages, Scriptcase can automatically create indexes for the application fields labels in the language files.

Generate indexes in language files with table field labels?

Overwrite language files indexes ?

Add Back

When synchronizing the tables, a yellow icon is displayed on the side of the updated tables, [See how to remove it](#) . In case of changing the structure of the tables, then you need to re-sync them again.

Synchronizing Applications

After doing all the desired configurations on the tables and fields, you need to sync the changes to the existing applications, so that the changes can be applied.

First, you need to select the tables that are synchronized.

SYNCHRONIZE APPLICATIONS

Select tables from the dictionary to update application fields and click Next to continue.

Next

<input checked="" type="checkbox"/> bar_code	{lang_tbl_bar_code}	1	Edit
<input checked="" type="checkbox"/> categories	{lang_tbl_categories}	1	Edit
<input type="checkbox"/> city	{lang_tbl_city}	1	Edit

Delete Save

Next, you can define what happens and which applications suffer the modifications — a list of applications that received the changes based on the tables that they used.

SYNCHRONIZE APPLICATIONS

Applications

Table: bar_code

grid51

Table: categories

ctr03_2

ctr04_2

form10

form13_1

form26

form29

Properties

Label

Size

Required

Minimum size

Max value

Help

Data type

Decimals

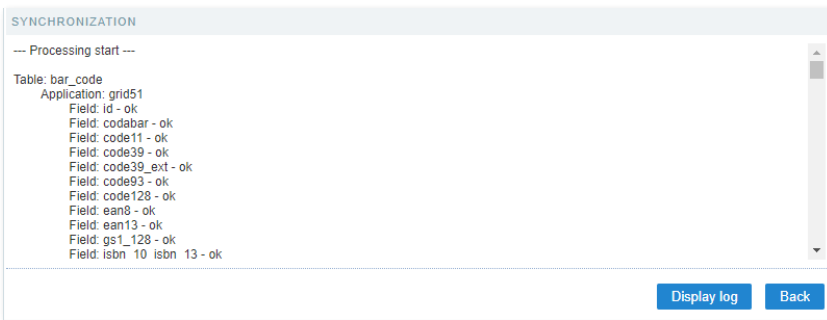
Maximum size

Min value

Mask

Synchronize Back

When clicking on synchronizing, when finishing the process, you see a summary of the procedure done.



Clicking to show the log, you can view all the modifications done, where it displays the previous state (blue) and the result of the modification (red)

Table: bar_code
Application: grid51

Field	Label	Label (search)
id	Before	id
	After	{lang_bar_code_fid_id}
codabar	Before	codabar
	After	{lang_bar_code_fid_codabar}
code11	Before	code11
	After	{lang_bar_code_fid_code11}

Clicking on **Back** You return to the Page of the definition of modifications.

To finish the process, click on close.

Editing the Data Dictionary Properties

On the top part, we can view the dictionary properties, informed when created.

Edit Data Dictionary: webhelp_dictionary

Description

Use schema before the table

Convert tables/fields to

Connection

Associate dictionary with connection

Edit Reload Close

You have three possible actions, Edit, Reload, and Close.

- edit - Allows you to change the properties data of the dictionary, like the modification of the connection used to associated with a repository, for example.

Edit Data Dictionary: webhelp_dictionary

Description

Use schema before the table

Convert tables/fields to

Connection

Associate dictionary with connection

Update Cancel

- reload - Updates all the information of the Data Dictionary, like the status of the table, for example.
- close - Closes the current data dictionary, and you go to the list of the dictionary.

Editing Indexes

To edit the created indexes in the repository, access **Locales > Application Language**.

On the side panel, it is possible to view all the lang from ScriptCase. The langs referring to the data dictionary is in the folder **Project Messages**.

Index	Portuguese/Português (Brasil)	Spanish/Español	English
lang_balance	Saldo	Saldo	Balance
lang_birth	Nascimento	Fecha de nacimiento	Birth date
lang_code	Código	Código	Code
lang_contact	Contato	Contacto	Contact
lang_name	Nome	Nombre	Name
lang_pass	Senha	Contraseña	Password

In this article, you learn how to edit various application at the same time. Using the Express Edit tool, you can define values for various common attributes for these applications. It is useful specially to add updates to your applications.

Only some settings are available within the Express Edit tool.

This tool can be accessed from the menu **Tools > Express Edit**.

Select Application

Firstly, You need to inform which applications You want to edit.

Express Editing

All
 Select Applications

All

You edit all the applications in the project, with no exceptions.

Select Applications

All the applications existing in the project are listed so that you can select the applications that you want to edit.

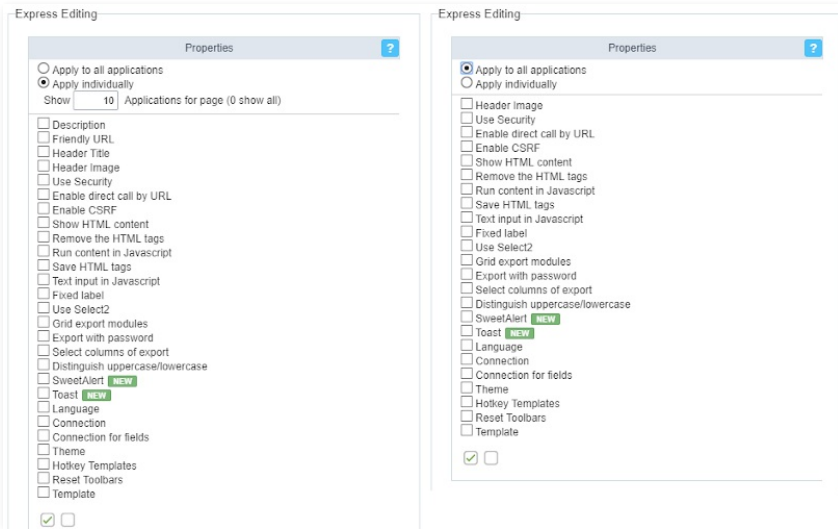
Applications ?

Visualization Per Type Per Folder

Tab		
<input checked="" type="checkbox"/> tab01_1	<input checked="" type="checkbox"/> tab02	<input checked="" type="checkbox"/> tab03
<input checked="" type="checkbox"/> tabs_v9	<input type="checkbox"/>	<input type="checkbox"/>
Blank		
<input type="checkbox"/> sample_blank_application	<input type="checkbox"/>	<input type="checkbox"/>
Calendar		
<input type="checkbox"/> calendar01	<input type="checkbox"/> calendar02	<input type="checkbox"/> calendar03
<input checked="" type="checkbox"/> calendar_color_event	<input type="checkbox"/>	<input type="checkbox"/>
Chart		
<input type="checkbox"/> chart05	<input type="checkbox"/> chart_54_1	<input type="checkbox"/> chart_54_2
<input type="checkbox"/> chart_54_3	<input type="checkbox"/> chart_54_4	<input type="checkbox"/> chart_area
<input type="checkbox"/> chart_area_sales	<input type="checkbox"/> chart_bars_2D	<input type="checkbox"/> chart_bars_3D
<input type="checkbox"/> chart_bubble	<input type="checkbox"/> chart_columns_2D	<input type="checkbox"/> chart_columns_3D
<input type="checkbox"/> chart_donut	<input type="checkbox"/> chart_employee_by_all	<input type="checkbox"/> chart_gauge
<input type="checkbox"/> chart_gauge81	<input type="checkbox"/> chart_line	<input type="checkbox"/> chart_line_step
<input type="checkbox"/> chart_multi_series	<input type="checkbox"/> chart_orders_by_employee	<input type="checkbox"/> chart_pie_2d
<input type="checkbox"/> chart_pie_3d	<input type="checkbox"/> chart_pizza	<input type="checkbox"/> chart_pyramid
<input type="checkbox"/> chart_pyramid81	<input type="checkbox"/> chart_scatter	<input type="checkbox"/> chart_spline
<input type="checkbox"/> chart_stacked_bar_3D	<input type="checkbox"/> chart_top_region_orders	<input type="checkbox"/> chart_work_employees
<input type="checkbox"/> combination_chart	<input type="checkbox"/> sample_chart	<input type="checkbox"/> sample_chart01
<input type="checkbox"/> sample_chart02	<input type="checkbox"/> sample_chart03	<input type="checkbox"/> sample_chart04
<input type="checkbox"/> tab03_2	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Apply Modifications

After selecting the applications, you define the modifications that you want for those applications. On this page, You can define how the modifications are applied.



Applying To All The Applications

This option allows applying the same configurations, to all the selected applications, automatically.

Header Image

Define the image for the header of the applications.

Use Security

Define the use of security on the applications of the project.

Enable direct call by URL

Allow the applications to be called directly by the URL.

Enable CSRF

It enables "Cross-Site Request Forgery" protection. (This option only applies for the Control and Form applications)

Show HTML content

Defines if the HTML code stored in the database is interpreted or not by the browser. (This option only applies for the Grid application)

Remove the HTML tags

Defines if the HTML code stored in the database will have the HTML tags removed for display.

Run content in JavaScript

Defines if the JavaScript code stored in the database runs on the browser.

Save HTML tags

Stores the HTML tags informed within the Application field to the database. (This option only applies for the Form application)

Text input in JavaScript

Stores the JavaScript informed within the Application field to the database. (This option only applies for the Form application)

Fixed Label

It allows you to transform the Fields labels in static labels when you scroll down the page.

Use Select2

It allows you to apply the Select2 as default in the applications.

Grid export modules

It allows you to select the Grid Modules in the exportation.

Export with Password

It allows defining a password in the Applications exportation.

Select columns of Export

It allows selecting the desired application columns in the exportation.

Distinguish uppercase/loYourcase

It allows the Scriptcase to distinguish the letters uppercase and loYourcase.

SYouetAlert

It allows using the alerts messages with SYouetAlert components.

Toast

It allows that all Scriptcase error messages use Toast error messages.

Language

It applies the selected language for all the selected applications in the project.

Connection

It applies the desired connection (Connections created in the "New Connection" module) for the applications.

Connection for fields

It applies the desired connection (Connections created in the "New Connection" module) for the fields in the applications.

Theme

It includes the desired Theme (Only the ones included in the Project properties) to all the applications.

Hotkey Templates

it allows configuring the hotkey template for all applications.

Reset Toolbar

It allows resetting the application toolbar settings.

Template

Change the Header and Footer template of the applications.

After selecting the modifications that you want to do for all the applications, you see a list with all the selected attributes.

Apply Individually

This option allows you to apply the modifications individually for each application selected. This option contains not only the previous attributes, but also three exclusive attributes.

Description

It applies a description individually for the selected applications.

Friendly URL

It applies a friendly URL individually for each selected application.

Title

It applies the title individually for each application.

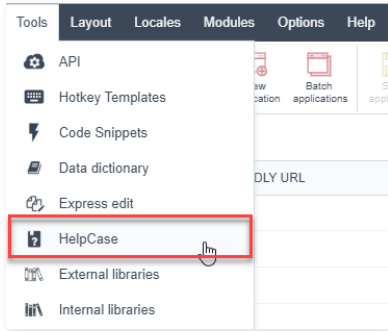
After selecting the modifications individually for all the select applications, It displays a list with all the selected attributes listed for modification.

When you apply the modifications, It displays a list of the applications that Youre modified.

Modified Applications		
tab01_1	tab02	tab03
tabs_v9		

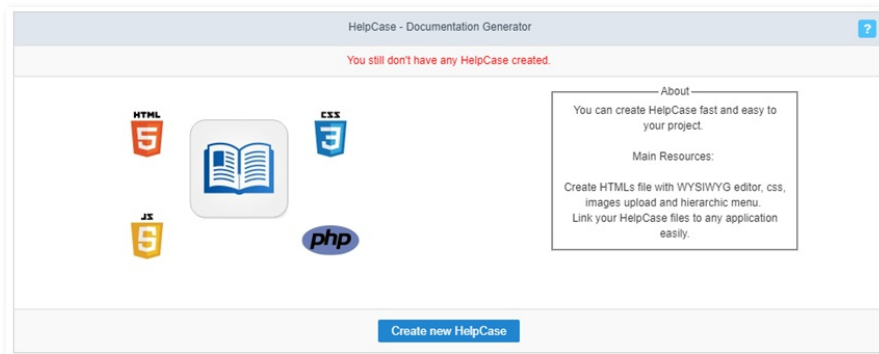
In this article, you learn how to create the documentation for your project using this integrated tool: HelpCase.

You find the HelpCase from the menu at **Tools > HelpCase**.

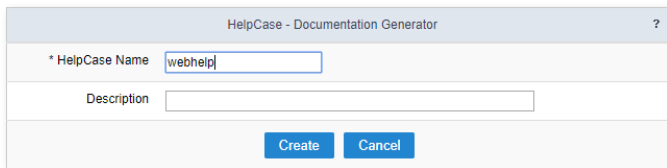


How to Create a HelpCase

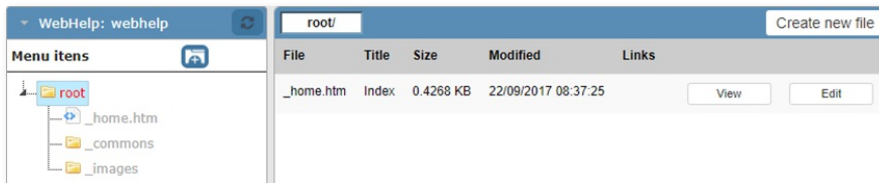
The HelpCase creates HTML files with WYSIWYG editor, CSS, images upload, and hierarchic menu. You can also link HelpCase files to the applications you create within Scriptcase.



The first step to creating a document using the HelpCase is to enter a name and description.



After creating the HelpCase, you need to generate the content of the documented manual of the functionalities and description of the project.



On the left side, in the block 'WebHelp,' you see the folder structure. Still, in this block, we can create the folders of the documentation.

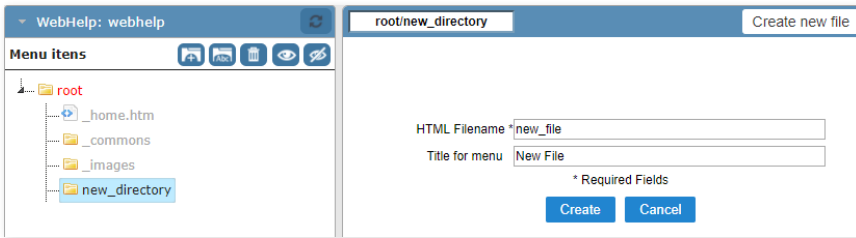
Select the root or any other folder on the side panel and click on the folder icon to create a folder in the select directory.

Inform the directory name (Special characters are not acceptable) and the display name in the documentation menu.



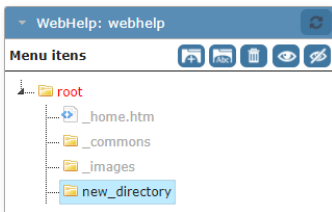
To create .html files, where you place your content, you need to select the directory where you place the files.

Inform the name of the file (Special characters are not acceptable) and the display name in the documentation menu.



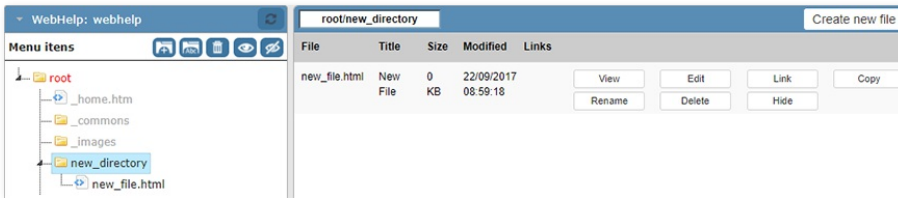
Editing the HelpCase

When you select the folder, you see some options.



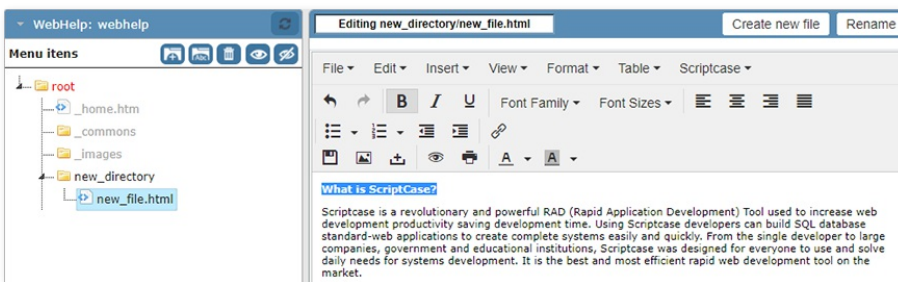
- **con (Folder with letters)** - Allows you to rename the archives and the display name for the menu.
- **con (Recycle Bin)** - Deletes the directory with all its content.
- **con (Eye)** - View the directory in the documentation menu.
- **con (Slashed Eye)** - Disable the view of the directory in the documentation menu.

After created, the files are displayed on the list when clicking on the created directory. Details about the files can be viewed, like the display title on the menu and its actual size, also other options like rename and delete.



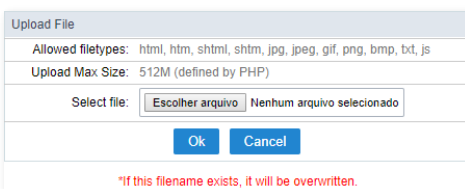
- **iew** - View the content of the file.
- **dit** - Open the file to place your content in it.
- **ink** - Create a link between the file and the applications selected.
- **opy** - Creates a copy of the file.
- **ename** - Rename the file and it is display menu.
- **elete** - Deletes the chosen file.

After the creation or clicking on edit a file, you can insert the content that will be displayed for the end user.



Upload

You can upload already created content or images to use in the manual. The upload of these files, can be done by clicking on the upload button, on the upload page you will see the supported formats and size limit of the files.



General Settings

Settings

Allows you to setup some things, like the header display and the search tab.

Settings	Value	Description
Use the new Helpcase generator	Yes	It sets the style of the generated manual. The old helpcase settings does not support the new customization options. So once you switch to the new Helpcase you will overwrite your current style sheet and lose the changes you have made previously.
Display header	<input checked="" type="radio"/> Enabled <input type="radio"/> Disabled	It enables or disables the header display of the generated manual.
Display home button	<input checked="" type="radio"/> Enabled <input type="radio"/> Disabled	It enables or disables the Home button display in the generated manual.
Display search tab	<input checked="" type="radio"/> Enabled <input type="radio"/> Disabled	It enables or disables the search tab display of the generated manual.
Display breadcrumb navigation	<input type="radio"/> Enabled <input checked="" type="radio"/> Disabled	Enables or disables the function to show the breadcrumb navigation inside the manual
Navigation anchors	<input type="text"/>	Defines after which HTML tag the breadcrumb navigation will be created. The default is after the body tag, e.g. h1

Initial Page

Allows to write content of the initial page of the documentation. The modification can be done also when selecting the `_home.htm` file on the files panel.

View

Define a template to be used in the header of the manual, it is possible to edit or create new templates in the editor.

This editor can be accessed from the menu **Layout > HTML Templates**

Header

Allows to configure what will be displayed in the header.

Description	Value	Description
Title	WebHelp	It sets the HTML "title" attribute value of the generated manual.
Logo	scriptcase__NM__img__NM__sc_pure_trans.png	It sets a logo image for the manual
Background color	<input type="color"/>	It sets the background color of the generated manual, the default color is #333333
LIN1_COL1	Logo	It defines the value for header variables.
LIN1_COL2	<input type="text"/>	It defines the value for header variables.
LIN1_COL3	<input type="text"/>	It defines the value for header variables.

CSS

Allows to modify the HelpCase CSS.

```

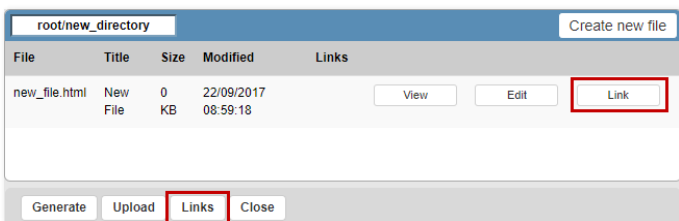
1 body{
2   background-color:#333;
3   margin:0;
4 }
5
6 a:hover { background: none; }
7
8 .btn_home {
9   position:absolute;
10  left:230px;
11  top:9px;
12 }
13
14 #abas{
15  font-family: Tahoma, Verdana, Geneva, Arial, Helvetica, sans-serif;
16  font-size: 13px; color:#000000;
17  vertical-align:text-top;
18 }
19 #id_bodyContent{
20  width:100%;
21  height:100%;
22  border-width:0;
23 }

```

After creating all the content and the configuration of the layout, you need to associate the create files with the applications, so that the end user can have access to the information that he needs in a quick way.

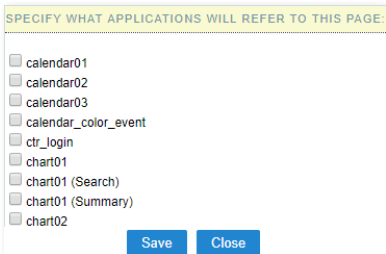
Associating Applications

You can start associating all the applications at once by clicking on **links**, or clicking on **link**, where you can associate only the chosen file to various applications.



Links

On this option, you will see a list of the applications, where you should select the applications that will be associated.



Link

Clicking on **link**, you will be presented a list of all the applications, in this case you need to select the files that will be associated for each application.

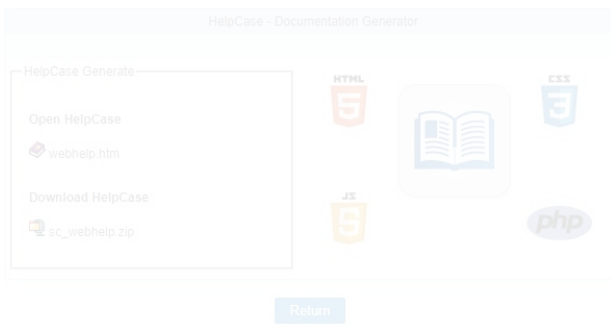
APPLICATION	DIRECTORY	DT. CREATING	
calendar01	root/Calendar	14/07/10 17:15	new_directory/new_file.html
calendar02	root/Calendar	19/07/10 18:43	
calendar03	root/Calendar	29/10/10 17:32	
calendar_color_event	root/Calendar	25/07/14 00:11	
chart01	root/Grid_Applications/Charts	25/01/13 10:02	
			Search: <input type="text"/>
			Summary: <input type="text"/>

Generating Manual

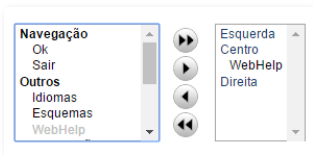
Generating the manual is the last step of its creation, after doing all the configurations, click on **generate**.



After generating the manual, you can download or access it.



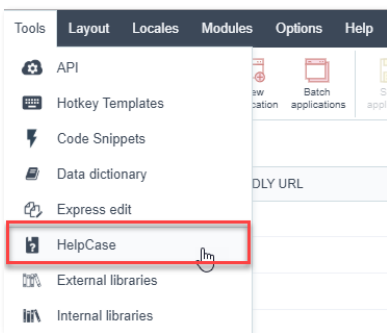
For you to have access to the manual from the applications, you need to click on the button **WebHelp** on the toolbar of the applications that have a link to the manual.



title: HelpCase layout: landing —

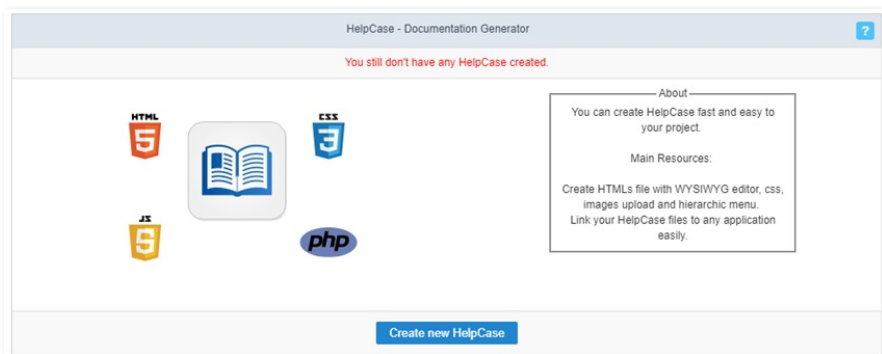
In this article, you learn how to create the documentation for your project using this integrated tool: HelpCase.

You find the HelpCase from the menu at **Tools > HelpCase**.

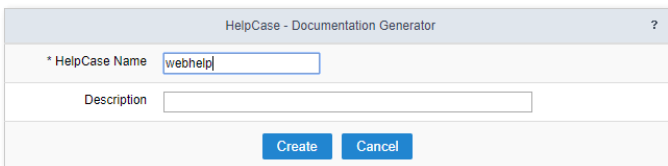


How to Create a HelpCase

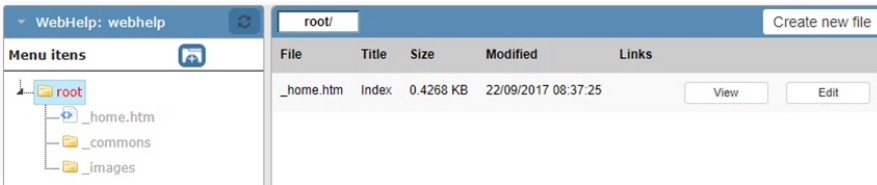
The HelpCase creates HTML files with WYSIWYG editor, CSS, images upload, and hierarchic menu. You can also link HelpCase files to the applications you create within Scriptcase.



The first step to creating a document using the HelpCase is to enter a name and description.



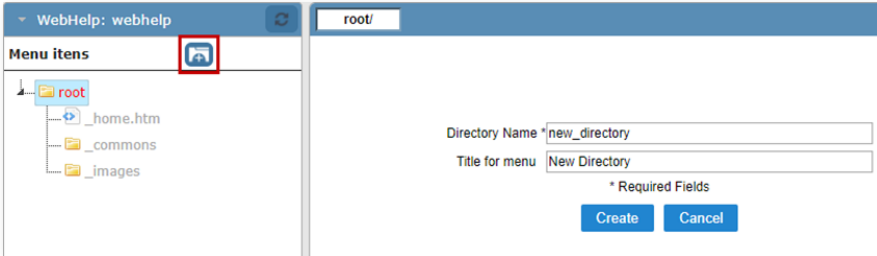
After creating the HelpCase, you need to generate the content of the documented manual of the functionalities and description of the project.



On the left side, in the block 'WebHelp', we will see the folder structure. Still in this block we can create the folders of the documentation.

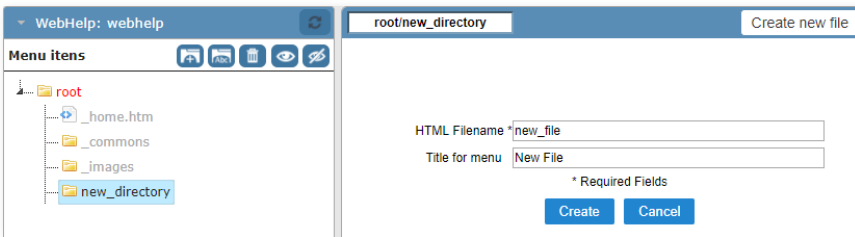
Select the root or any other folder on the side panel and click on the folder icon. This way a folder will be created in the select directory.

Inform the directory name (Special characters are not acceptable) and the display name in the documentation menu.



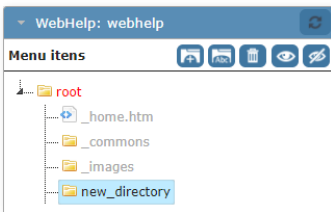
To create .html files, where you will place your content, you need to select the directory where you will place the files in.

Inform the name of the file (Special characters are not acceptable) and the display name in the documentation menu.



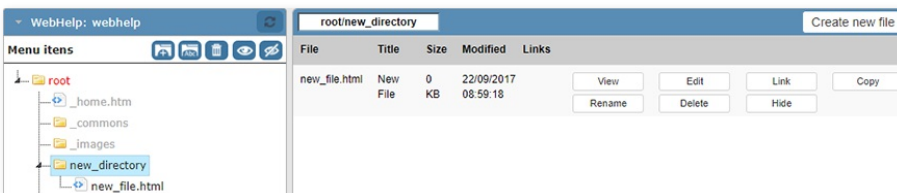
Editing the helpcase

Selecting the folder that you will create, you will view some new option with the create button.



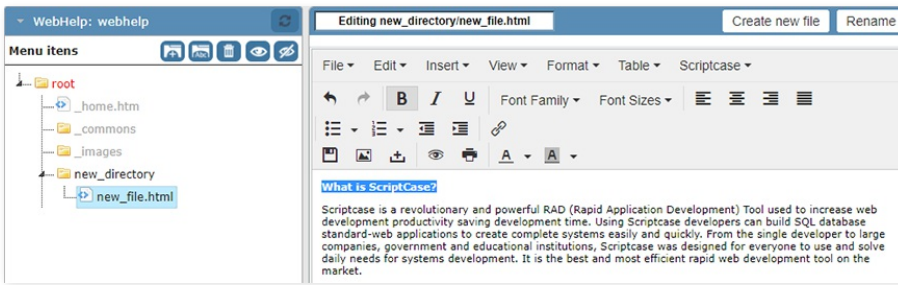
- **con (Folder with letters)** - Allows you to rename the archives and the display name for the menu.
- **con (Recycle Bin)** - Deletes the directory with all its content.
- **con (Eye)** - View the directory in the documentation menu.
- **con (Slashed Eye)** - Disable the view of the directory in the documentation menu.

After created, the files will be displayed on the list, when clicking on the directory that were created. Details about the files can be viewed, like the display title on the menu and its actual size, also other options like rename and delete.



- **iew** - View the content of the file.
- **dit** - Open the file to place your content in it.
- **ink** - Create a link between the file and the applications selected.
- **opy** - Creates a copy of the file.
- **ename** - Rename the file and it is display menu.
- **elete** - Deletes the chosen file.

After the creation or clicking on edit a file, you can insert the content that will be displayed for the end user.



Upload

You can upload already created content or images to use in the manual. The upload of these files, can be done by clicking on the upload button, on the upload page you will see the supported formats and size limit of the files.

Upload File	
Allowed filetypes:	html, htm, shtml, shtm, jpg, jpeg, gif, png, bmp, txt, js
Upload Max Size:	512M (defined by PHP)
Select file:	<input type="button" value="Escolher arquivo"/> Nenhum arquivo selecionado
<input type="button" value="Ok"/> <input type="button" value="Cancel"/>	
*If this filename exists, it will be overwritten.	

General Settings

Settings

Allows you to setup some things, like the header display and the search tab.

Settings: Basic			
Settings	Value	Description	
Use the new Helpcase generator	<input checked="" type="checkbox"/> Yes	It sets the style of the generated manual. The old helpcase settings does not support the new customization options. So once you switch to the new Helpcase you will overwrite your current style sheet and lose the changes you have made previously.	
Display header	<input checked="" type="radio"/> Enabled <input type="radio"/> Disabled	It enables or disables the header display of the generated manual.	
Display home button	<input checked="" type="radio"/> Enabled <input type="radio"/> Disabled	It enables or disables the Home button display in the generated manual.	
Display search tab	<input checked="" type="radio"/> Enabled <input type="radio"/> Disabled	It enables or disables the search tab display of the generated manual.	
Display breadcrumb navigation	<input type="radio"/> Enabled <input checked="" type="radio"/> Disabled	Enables or disables the function to show the breadcrumb navigation inside the manual	
Navigation anchors	<input type="text"/>	Defines after which HTML tag the breadcrumb navigation will be created. The default is after the body tag, e.g. h1	

Initial Page

Allows to write content of the initial page of the documentation. The modification can be done also when selecting the `_home.htm` file on the files panel.

View


Define a template to be used in the header of the manual, it is possible to edit or create new templates in the editor.

This editor can be accessed from the menu **Layout > HTML Templates**

Layout: View	
Settings	Value
Header template	default <input type="button" value="Template editor"/>
<input type="button" value="Save"/>	

Header

Allows to configure what will be displayed in the header.

Settings: Header		
Description	Value	Description
Title	<input type="text" value="WebHelp"/>	It sets the HTML "title" attribute value of the generated manual.
Logo	<input type="text" value="scriptcase_NM_img_NM_sc_pure_trans.png"/> 	It sets a logo image for the manual
Background color	<input type="text" value="#333333"/>	It sets the background color of the generated manual, the default color is #333333
LIN1_COL1	<input type="text" value="Logo"/>	It defines the value for header variables.
LIN1_COL2	<input type="text"/>	It defines the value for header variables.
LIN1_COL3	<input type="text"/>	It defines the value for header variables.

CSS

Allows to modify the HelpCase CSS.

```

1 body{
2   background-color:#333;
3   margin:0;
4 }
5
6 a:hover { background: none; }
7
8 .btn_home {
9   position:absolute;
10  left:230px;
11  top:9px;
12 }
13
14 #abas{
15  font-family: Tahoma, Verdana, Geneva, Arial, Helvetica, sans-serif;
16  font-size: 13px; color:#000000;
17  vertical-align:text-top;
18 }
19 #id_bodyContent{
20  width:100%;
21  height:100%;
22  border-width:0;
23 }

```

After creating all the content and the configuration of the layout, you need to associate the create files with the applications, so that the end user can have access to the information that he needs in a quick way.

Associating Applications

You can start associating all the applications at once by clicking on **links**, or clicking on **link**, where you can associate only the chosen file to various applications.

root/new_directory					Create new file	
File	Title	Size	Modified	Links		
new_file.html	New File	0 KB	22/09/2017 08:59:18	<input type="button" value="View"/>	<input type="button" value="Edit"/>	<input type="button" value="Link"/>

Links

On this option, you will see a list of the applications, where you should select the applications that will be associated.

SPECIFY WHAT APPLICATIONS WILL REFER TO THIS PAGE:

- calendar01
- calendar02
- calendar03
- calendar_color_event
- ctr_login
- chart01
- chart01 (Search)
- chart01 (Summary)
- chart02

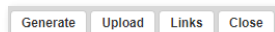
Link

Clicking on **link**, you will be presented a list of all the applications, in this case you need to select the files that will be associated for each application.

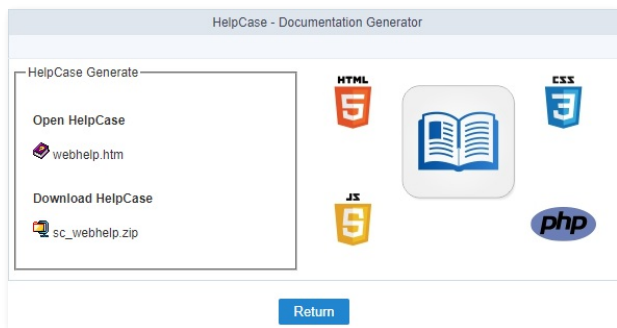
APPLICATION	DIRECTORY	DT. CREATING	
calendar01	root/Calendar	14/07/10 17:15	new_directory/new_file.html ▼
calendar02	root/Calendar	19/07/10 18:43	▼
calendar03	root/Calendar	29/10/10 17:32	▼
calendar_color_event	root/Calendar	25/07/14 00:11	▼
chart01	root/Grid_Applications/Charts	25/01/13 10:02	▼
			Search: ▼
			Summary: ▼

Generating Manual

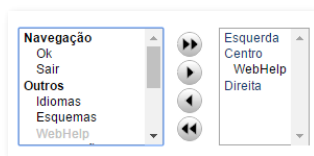
Generating the manual is the last step of its creation, after doing all the configurations, click on **generate**.



After generating the manual, you can download or access it.



For you to have access to the manual from the applications, you need to click on the button **WebHelp** on the toolbar of the applications that have a link to the manual.



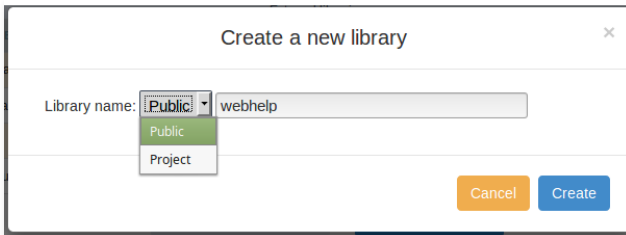
This feature allows to import complete libraries and create files of any extension (.JS, .CSS, .PHP...) that can be used in your projects, making it easier to reuse some of the source code.

This feature is available in the menu **Tools > External Libraries**, where you can manage the libraries of your project.

To use this functionality, you need to use the macro, `sc_url_library()` and the `sc_include_library()` (Only for PHP files) to import the library to the applications.

Creating a new library

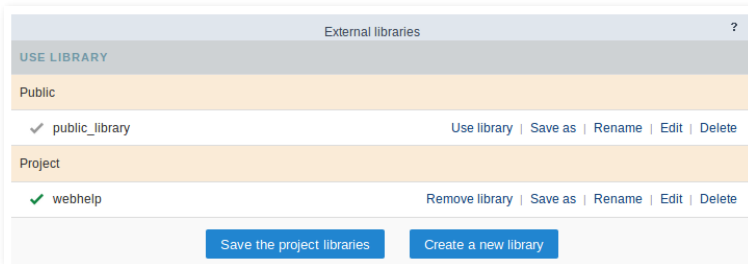
To create a library, click on **Create a new library**. Next, inform the library name and the access level of the library.



- **ublic** - Available for all the ScriptCase projects.
- **roject** - Available only for the project it was created in.

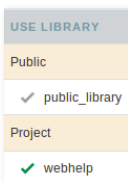
Managing The Libraries

All the public and project libraries are listed here. This page allows you to select the libraries that will be part of the project, delete the unnecessary libraries, rename or edit them.

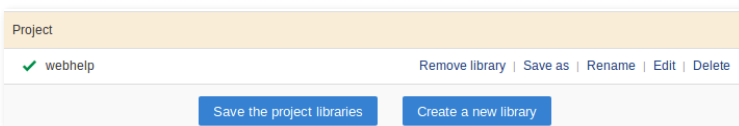


Using a Library

This option defines a library that will be used in the current project. The enabled libraries, for this project, can be differentiated by the checkmark next to the name of the library.



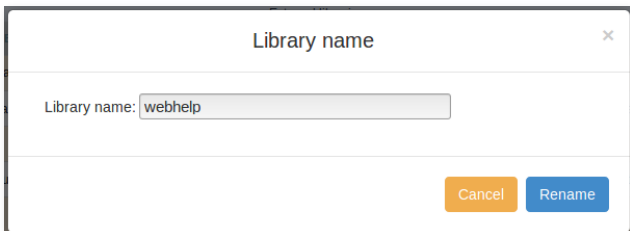
Selecting a library to be used in the project, the icon changes color when you click on the button "Save the library to the project" and enabled, you need to click on the button to save the modifications.



Rename

To rename a library just click on "rename", and then inform the new name for the library.

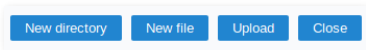
To rename a library, you will need to change all the includes of the library in the events of the applications.



Editing a Library

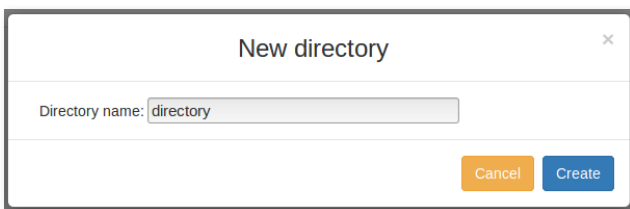
Clicking on "Editing", you can create or upload files to the libraries. You can create files in a folder, with the goal to organize your files.

On Footer of the page, you can view the options to create or upload the files.



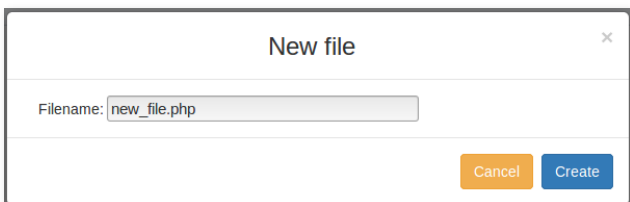
New Directory

To create a directory, click on "New Directory" and inform the name of the directory.



New File

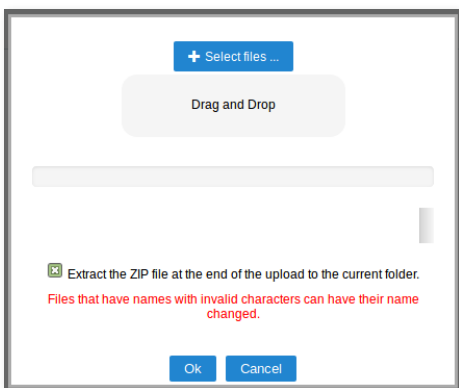
To create a file, click on "New File" and inform the name and extension (it is indispensable to inform the file extension).



Upload Files

This tool, allows to upload files and external libraries and used them in the project.

Clicking on upload, select the desired file.



Delete

Delete the libraries and all the files contained in them.

Before deleting the library, be certain that it is not being used, you will not be able to recover deleted libraries.

Importing libraries into the applications

Like mentioned before, to use the libraries, you need to use the macro `sc_url_library()` and the `sc_include_library()` (Only for PHP files).

The macro `sc_url_library()` returns the path to the file in the library and the `sc_include_library()` is used to include php files into applications from the external library:

```
onScriptInit
1 ?>
2 <link rel="stylesheet" type="text/css" href="<?php echo sc_url_library('prj', 'webhelp',
3 <?PHP
   'directory/style.css');?>">
```

This example changes the background color of a grid application, if the code in the style.css file

```
1 .scGridPage
2 {
3     background-color: #1c6f16 !important;
4 }
```

This tool allows to create or upload PHP routines that can be reused in various applications of the project or other projects, depending on the level of access that they were saved.

There are three types of access levels that can be used.

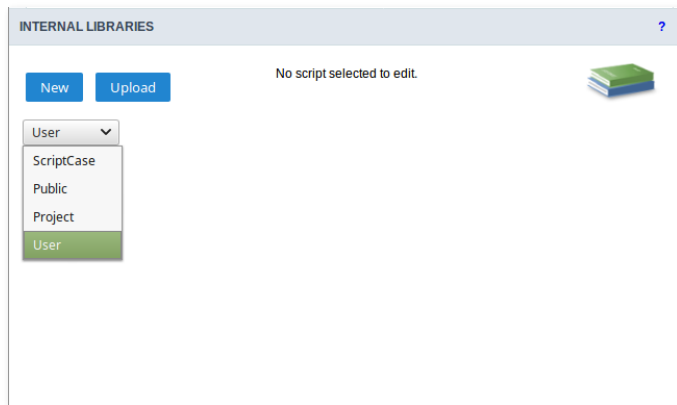
- **ser** - Only for the user that created the script will have access.
- **roject** - All the users linked to the project will have access to the script.
- **ublic** - All the users from ScriptCase will have access to the script.

You can't create or modify scriptcase that are in the Scriptcase Level, the scripts from this level are available to be used but nor modified.

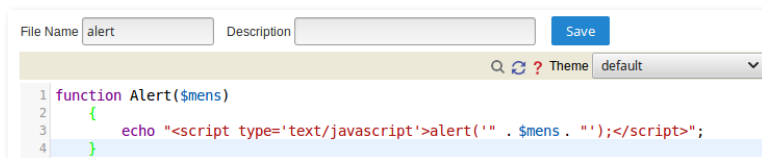
Creating A Script

The item **Internal Libraries** are accessed from the tools menu.

Select the level that the script is being saved, in this example the user level was used.

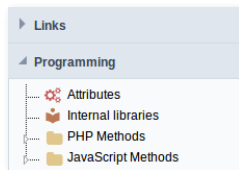


Next, inform a name for the file, place your code and save it.



Using The Script

To use the Scripts, firstly you need to enable them in the applications that you want to use.



All the libraries that you have access will be listed, just select the the ones you desire to enable and use them.



To use the libraries, you need to import them by using the macro **sc_include**.

Example of using the macro: `sc_include("File", "Origin");`



Tool that allows the creation of a list of tasks for the users registered in the same ScriptCase, with a progress control for the task.

A message that will always be displayed when the users that possess a task pending login.

To access the feature, go to the menu **Tools > To-Do List**

Accessing it, you will see all the registered tasks with the information about each one of them.

Task list	Application	Owner	Responsible	Percentage done	Deadline	Options
Task 3 - Project		admin	admin	0%	01/31/2018 00:00	Edit Delete
Task 2		admin	admin	0%	01/31/2018 00:00	Edit Delete
Task 1		admin	admin	0%	01/31/2018 00:00	Edit Delete

Creating A Task

The tasks can be created in four levels.

- **ublic List** - All the ScriptCase users can view the tasks of the this level.
- **y List** - Only the Creator and the user responsible for the task can view.
- **roject: Public List** - All the users of ScriptCase, linked to the project that the task was created can view them. (This option is available only when a project is opened)
- **roject: My List** - Only the Creator and the Responsible user can view this task, they need to have access to the project where the task was created. (This option is available only when a project is opened)

Firstly, you need to select where you are going to create the task. Selecting the folder, the "New Task" button will be enabled.

NEW TASK

*Task list

Percentage done

*Deadline

Responsible

Who can change

Application

Changes in the dashboard application

Next, after clicking on "New Task", you need to fill up the required fields.

- **ask list** - Name of the task, which will be displayed in the Task list.
- **ercentage done** - Inform the percentage of the task conclusion.
- **eadline** - Define the date and time that the task need to be concluded.
- **esponsible** - Define which of the ScriptCase users are responsible for the task.
- **ho can change** - Defines who can modify the current task:
 - **wner** - Only the user that created the task.
 - **esponsible** - Only the user responsible for the task.
 - **ll** - All the ScriptCase users can modify the task.
- **pplication** - Inform the application of the project that the responsible user needs to edit. (This option will only be available for task created in a project.)
- **escription** - Task Description.

Task Status

This tool is used to inform the status of the task, using colors to symbolize the progress.

- **pen tasks** - Tasks recently created within the deadline, are represented in yellow.
- **ompleted tasks** - Tasks that finished completely, are represented in blue.
- **xpired tasks** - Tasks that passed the deadline, are represented in red.

All

Public List

My List

Project: Public List

Project: My List

Open tasks

Completed tasks

Expired tasks

Task list

New Task

✓ Task 3 - Project

Task 1

Task 2

The message system allows the communication between the registered developers on the same ScriptCase.

You can access this feature. **Tools > Messages**.

New Message

After clicking on the New Message button, you need to select the user that is going to receive the message.

One message can be sent to many users.

TO

[Select all]

admin

After typing the message, click on send.

TO

All selected

SUBJECT

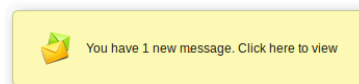
Creating webhelp

Scriptcase's message sending example

Send
Cancel

Inbox

Receiving the message, the users will be notified by an alert.



You can click on the notification bell or access **Tools > Message** to view the received messages.

Accessing the inbox where all the received messages are stored and ordered by the date.

New Message		Subject	Sent in
Inbox		<input type="checkbox"/> Creating webhelp	01/11/18 14:32
Outbox			

Outbox

The sent messages are stored in the outbox.

By sending a message to various users, it will be registered a message for each user in the outbox.

Subject	Sent in	From	To
<input type="checkbox"/> Creating webhelp	01/11/18 14:32	admin	admin
<input type="checkbox"/> Creating webhelp	01/11/18 14:32	admin	admin

Read and Reply

Click on the message that you want to read.

From	admin
To	admin
Sent in	01/11/18 14:32
Subject	Creating webhelp

Scriptcase's message sending example

Reply Close

When reading a message, you can reply the received message.

TO
All selected

SUBJECT
Re: Creating webhelp

[quote]Scriptcase's message sending example[/quote]

Send Cancel

Received Response

In the Outbox, a responded message can be identified with a Re: before the subject

Subject	Sent in
<input type="checkbox"/> Re: Creating webhelp	01/11/18 14:56
<input type="checkbox"/> Creating webhelp	01/11/18 14:32
<input type="checkbox"/> Creating webhelp	01/11/18 14:32

The original message, will be highlighted in the received response.

From	admin
To	admin
Sent in	01/11/18 14:56
Subject	Re: Creating webhelp




Scriptcase's message sending example


Delete Messages

You can delete the messages individually by clicking on the delete link.

To	Read in	Options
admin	01/11/18 14:56	View Delete
admin	01/11/18 14:53	View Delete
admin	01/11/18 14:53	View Delete

You can also delete various messages. Select all the messages that you want to delete and then click on the delete button.

Subject	Sent in
 Re: Creating webhelp	01/11/18 14:56
 Creating webhelp	01/11/18 14:32
 Creating webhelp	01/11/18 14:32

 Delete

Project Converters

The Project Converter allows to convert the projects developed in previous versions of Scriptcase to the current version of Scriptcase.

Basically it is an import of the projects from different versions of Scriptcase.

The process of conversion does not modify any of the information of your project.

To prevent eventual problems, it is recommend to create a backup of your projects.

The project's conversion works the same way for all Scriptcase versions.

Coping Scriptcase

If your Scriptcase is not installed in the same place of your Scriptcase 9 you need to copy the Scriptcase folder, of the previous version, nexto to the current ScriptCase folder that will receive the projects.

In this example:

- **criptcase 8 folder** - Old version of the tool.
- **criptcase 9 folder** - Version that will receive all the projects.

scriptcase	15/01/2018 16:07
scriptcase8	15/01/2018 16:15
favicon.ico	05/11/2013 12:25
index.php	05/11/2013 12:25
robots.txt	05/11/2013 12:25

Coping the Scriptcase 8 folder to the same place

Path to Scriptcase

In Scriptcase select the converter (compatible to the version that you want to convert), and inform the complete path to the folder of the old version.

If your Scriptcase 9 is installed in the same place of your old version you can inform the complete path where your old version is installed.

- **x1: /opt/NetMake/v9/wwwroot/scriptcase**
- **x2: C:\Program Files(x86)\NetMake\v81\wwwroot\scriptcase**

Migrating Scriptcase projects to version 9

SCRIPTCASE VERSION 8 PATH

Scriptcase 8 Path

Important!

- If the Scriptcase 8 is installed on the same computer you just inform the complete directory path, if not, you will need to copy the folder Scriptcase 8 to Scriptcase 9 wwwroot folder. Remember to change the folder name to scriptcase8, for example.
- The conversion process will copy and convert all projects, applications, schema, data dictionary, languages, and etc, to work fully in version 9. This process does not change in any moment the previous version projects.

Check out the tutorial for project conversion:

[Video](#) | [Tutorial](#) | [FAQ](#) | [Docs](#)

Informing the path to the Scriptcase 8

Converting the projects from your old version

In this screen will appear all projects from your old version.

- **onvert**: Will convert the projects from your old version.
- **ot Convert**: Will not convert the projects from your old version.

*If is not the first time that you are converting, may appear a option called **Overwrite**. If you convert with this option selected will overwrite your project with the old one.*

MIGRATING SCRIPTCASE PROJECTS TO VERSION 9

EXISTING PROJECTS ONLY IN V8.

TODO Convert Not Convert

Reverse

Convert

Exit

Selecting the projects that will be converted

Next, you will view a log of everything that has been converted. With this the selected projects are already in the current ScriptCase and ready for use.

Migrating Scriptcase projects to version 9

SUMMARY

Migrating ScriptCase tables to V9

- Cleaning up Scriptcase V9 database
- Migrating table - sc_tbprj
- Migrating table - sc_tbusu
- Migrating table - sc_tbapl
- Migrating table - sc_tbevt
- Migrating table - sc_tbconex
- Migrating table - sc_tbrep
- Migrating table - sc_tbrep_tables
- Migrating table - sc_tbrep_fields
- Migrating table - sc_tversao
- Migrating table - sc_tblog
- Migrating table - sc_ttodo

Converting Templates, Schemas and Buttons:

- Migrating file directory
- Migrating templates: Public
- Migrating templates: Projects
- Migrating templates: Users
- Migrating Color Schemes

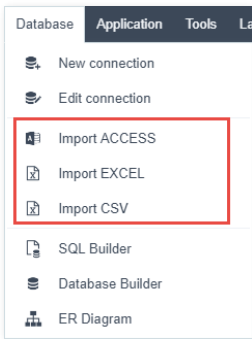
Migration finished with success!

Exit

Information showing the success

Allows you to convert one or more data sources to one structured database. You can convert the stored tables in .XLS, .CSV, .accdb and .mdb files to the following databases: **MySQL**, **PostgreSQL**, **SQLite** and **SQL Server**.

You can find all the options to import databases in the menu **Database**.



Access Import

Selecting Files

First, select the files that will be converted. You can inform the absolute path to the file, drag and drop the file or select the file in the option **Select Files**.

!m[Selecting an access file to convert the tables](#)

Configuring The Tables

After selecting the file, you will need to access the configuration page.

On this page, you will see the password field, that is only used with **Access** files that were selected and require a password to list the tables. In these cases, after informing the password, click on **list tables**.

The screenshot shows a form titled 'DATASOURCE SETTINGS'. It has a 'Password' input field and a text label 'Password required to open the Access file.' Below the input field is a blue button labeled 'List table'.

Still in the same page, you can view the tabs **Tables** and **Advanced**.

The screenshot shows the 'Tables' tab selected. It displays a list of tables from an Access database file. Each table has a checkbox and 'Edit | Preview' links.

Tables	Options
C:/Program Files/NetMake/v9/wwwroot/scriptcase/tmp/sc_tmp_upload_05h3q410cibflok12vcvb4t77/sc_finances.mdb	
<input checked="" type="checkbox"/> accounts	Edit Preview
<input checked="" type="checkbox"/> categories	Edit Preview
<input checked="" type="checkbox"/> planning_goals	Edit Preview
<input checked="" type="checkbox"/> planning_goals_type	Edit Preview
<input checked="" type="checkbox"/> sec_users	Edit Preview
<input checked="" type="checkbox"/> transaction_types	Edit Preview
<input checked="" type="checkbox"/> transactions_account	Edit Preview
<input checked="" type="checkbox"/> user_account	Edit Preview

Table Tabs

We can view a list of the tables that can be converted. Beside the name of the tables, are two options: Edit and Preview.

On the option **Edit**, you can modify the table structure that will be created, like the name, field size, and the data type and other options.

Table Definition: **accounts** Foreign Keys Unique Keys

Table Name:

Source			Target					
Field	SQL Type	Size	Field	SQL Type	Size	Not null	PK	AI
<input checked="" type="checkbox"/> account_id	integer	4	<input type="text" value="account_id"/>	integer	4	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> account_desc	varchar	45	<input type="text" value="account_desc"/>	varchar	45	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

in **Preview** you will see a preview of how the table will be imported.

Preview: **accounts**

account_id	account_desc
1	Wallet
2	Bank
3	Credit Card
4	Savings

[1 to 4 of 4]

Advanced Tab

On the "Advanced" tab the configurations import script for the table are available. You can configure the script that will be created for the generation of the tables of new database.

Tables: **Advanced**

Options

Add DROP TABLE before create the tables

Structure and Data

Recreate structure and data

Recreate structure


Recreate only data

- dd DROP TABLE - Adds a DROP TABLE IF EXISTS before the CREATE of the tables, so that if the tables exist with the same name in the database, they will be deleted first.
- ecreate the database structure - Where you will define the table structure and the data that will be imported.
- ecreate the structure - Where you will define only the structure the table that will be imported.
- ecreate data - Where you will define only the data of the tables that will be imported (In case the tables exist in the database with the same structure of the ones being imported)


Connection

On this step, we need to select the connection (Database) to where you will import the tables. You can select a connection existing in the project or create a new one.


CONNECTIONS




New connection




conn_mssql



conn_mysql

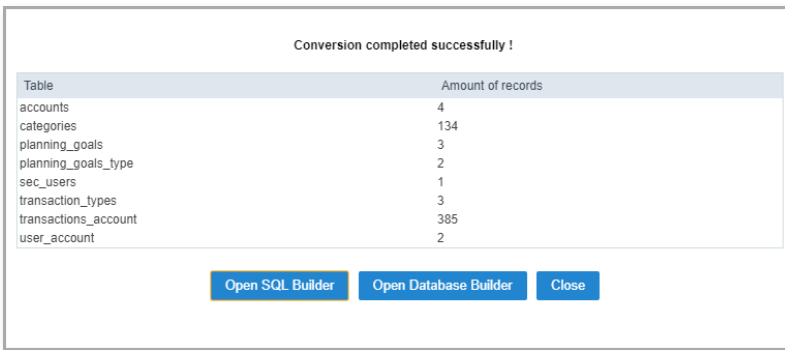


conn_mysql_1



routerboxdb

Next, you will be presented with a log of the tables that have been imported and the total of records inserted.



XLS Import

Selecting Files

First, select the files that will be converted. You can inform the absolute path to the file, drag and drop the file or select the file in the option **Select Files**.

[!mSelecting an access file to convert the tables](#)

Configuring The Tables

After selecting the file, you need to access the configuration page.

On this page, you need to inform first the line that the name of the columns are, this way all the information that is below will be considered a record of the table. If you do not inform the line, all the file content will be considered a record.

DATASOURCE SETTINGS

In which row the columns are

[List table](#)

Still in the same page, you can view the tabs **Tables** and **Advanced**.

Tables Advanced

Tables	Options
sample.xlsx	
<input checked="" type="checkbox"/> zpcp_prog	Edit Preview

[Back](#)
[Next](#)

You can view below the list of tables of the files, with the names of the tables and the name of the file. In the case of XLS or CSV files, the table names are obtained through the name of the tab in the file.

Table Tabs

We can view a list of the tables that can be converted. Beside the name of the tables, are two options: Edit and Preview.

On the option **Edit**, you can modify the table structure that will be created, like the name, field size, and the data type and other options.

Table Definition: **accounts** Foreign Keys Unique Keys

Table Name

Source			Target					
Field	SQL Type	Size	Field	SQL Type	Size	Not null	PK	AI
<input checked="" type="checkbox"/> account_id	integer	4	<input type="text" value="account_id"/>	integer ▼	<input style="width: 20px;" type="text" value="4"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> account_desc	varchar	45	<input type="text" value="account_desc"/>	varchar ▼	<input style="width: 20px;" type="text" value="45"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

[Cancel](#)
[Save](#)

in **Preview** you will see a preview of how the table will be imported.

Preview: **accounts**

account_id	account_desc
1	Wallet
2	Bank
3	Credit Card
4	Savings

[1 to 4 of 4]

[Close](#)

Advanced Tab

On the "Advanced" tab the configurations import script for the table are available. You can configure the script that will be created for the generation of the tables of new database.

Tables **Advanced**

Options

Add DROP TABLE before create the tables

Structure and Data

Recreate structure and data

Recreate structure

Recreate only data






[Back](#) [Next](#)

- dd DROP TABLE - Adds a DROP TABLE IF EXISTS before the CREATE of the tables, so that if the tables exist with the same name in the database, they will be deleted first.
- ecreate the database structure - Where you will define the table structure and the data that will be imported.
- ecreate the structure - Where you will define only the structure the table that will be imported.
- ecreate data - Where you will define only the data of the tables that will be imported (In case the tables exist in the database with the same structure of the ones being imported)

Connection

On this step, we need to select the connection (Database) to where you will import the tables. You can select a connection existing in the project or create a new one.

CONNECTIONS

New connection conn_mssql conn_mysql conn_mysql_1 routerboxdb

[Back](#) [Convert now](#)

Next, you will be presented with a log of the tables that have been imported and the total of records inserted.

Conversion completed successfully !

Table	Amount of records
accounts	4
categories	134
planning_goals	3
planning_goals_type	2
sec_users	1
transaction_types	3
transactions_account	385
user_account	2

[Open SQL Builder](#) [Open Database Builder](#) [Close](#)

CSV Import

Selecting Files

First, select the files that will be converted. You can inform the absolute path to the file, drag and drop the file or select the file in the option **Select Files**.

!mSelecting an access file to convert the tables

Configuring The Tables

After selecting the file, you need to access the configuration page.

On this page, you need to inform first the line that the name of the columns are, this way all the information that is below will be considered a record of the table. If you do not inform the line, all the file content will be considered a record.

DATASOURCE SETTINGS

In which row the columns are

[List table](#)

Still in the same page, you can view the tabs **Tables** and **Advanced**.

Tables **Advanced**

Tables	Options
sample.xlsx	
<input checked="" type="checkbox"/> zpcp_prog	Edit Preview

[Back](#) [Next](#)

You can view below the list of tables of the files, with the names of the tables and the name of the file. In the case of XLS or CSV files, the table names are obtained through the name of the tab in the file.

Table Tabs

We can view a list of the tables that can be converted. Beside the name of the tables, are two options: Edit and Preview.

On the option **Edit**, you can modify the table structure that will be created, like the name, field size, and the data type and other options.

Table Definition: **accounts** Foreign Keys Unique Keys

Table Name

Source			Target					
Field	SQL Type	Size	Field	SQL Type	Size	Not null	PK	AI
<input checked="" type="checkbox"/> account_id	integer	4	<input type="text" value="account_id"/>	integer	<input type="text" value="4"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> account_desc	varchar	45	<input type="text" value="account_desc"/>	varchar	<input type="text" value="45"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

[Cancel](#) [Save](#)

in **Preview** you will see a preview of how the table will be imported.

Preview: **accounts**

account_id	account_desc
1	Wallet
2	Bank
3	Credit Card
4	Savings

[1 to 4 of 4]

[Close](#)

Advanced Tab

On the "Advanced" tab the configurations import script for the table are available. You can configure the script that will be created for the generation of the tables of new database.

Tables **Advanced**

Options

Add DROP TABLE before create the tables

Structure and Data

Recreate structure and data

Recreate structure

Recreate only data






Back **Next**

- dd DROP TABLE - Adds a DROP TABLE IF EXISTS before the CREATE of the tables, so that if the tables exist with the same name in the database, they will be deleted first.
- ecreate the database structure - Where you will define the table structure and the data that will be imported.
- ecreate the structure - Where you will define only the structure the table that will be imported.
- ecreate data - Where you will define only the data of the tables that will be imported (In case the tables exist in the database with the same structure of the ones being imported)

Connection

On this step, we need to select the connection (Database) to where you will import the tables. You can select a connection existing in the project or create a new one.

CONNECTIONS

New connection conn_mssql conn_mysql conn_mysql_1 routerboxdb

Back **Convert now**

Next, you will be presented with a log of the tables that have been imported and the total of records inserted.

Conversion completed successfully !

Table	Amount of records
accounts	4
categories	134
planning_goals	3
planning_goals_type	2
sec_users	1
transaction_types	3
transactions_account	385
user_account	2

Open SQL Builder **Open Database Builder** **Close**

This tool helps on the construction of Queries in a Wizard form. These queries can be saved and used later.

Database

Any SQL can run directly from the **Run** tab that will be available after selecting the database.

Firstly you need to select the connection(Database) that you want to use.

Tables

All the existing tables in the database are listed in the **Tables** tab.

Select the desired tables to create the query.

You can select various tables, although they need to have a relationship between them.

Fields

Select the fields that will be part of the query. The fields can be ordered alphabetically (Ascendant) or in the order that they were created (Default)

It is possible to create an **alias** for the selected tables, this option can be found below the field list were also all the selected tables will be listed.

Joins

Defining the fields, it is necessary to configure the join. To do that, you need to select the tables that have a relationship, next you will select the type of join that will be used in the field **Type** then click on Add.

After clicking on add, click on **Confirm**, so that you can inform the fields that do the relationship between the tables.

Defining the fields, you will be redirected to the run tab.

To add a "WHERE" clause to the SQL, click on the tab **Conditions**

Conditions

You can add the "WHERE" clauses where the query was created.

Order By

Define the sorting of the query result display.

Run

This is where you can run the SQL commands. The result of the query configuration wizard will be displayed on this tab. You can run DML (Data Manipulation Language) commands here and also DDL (Definition Data Language) commands.

```

1 SELECT
2   employees.employeeid,
3   employees.lastname,
4   employees.firstname,
5   employees.title,
6   employees.titleofcourtesy,
7   employees.birthdate,
8   employees.hiredate,
9   employees.country,
10  employees.regionid,
11  employees.stateid,
12  employees.cityid
13 FROM
14  employeeterritories,
15  employees INNER JOIN employeeprojects ON employees.employeeid = employeeprojects.employeeid
16  INNER JOIN employeestate ON employees.employeeid = employeestate.employeeid
17 WHERE
18  (employees.employeeid > '5')
19 ORDER BY
20  employees.employeeid

```

Position: Ln 1, Ch 1 Total: Ln 20, Ch 518

Record limit per page (100 max)

[Run](#)

You can define a total of records per page.


employeeid	lastname	firstname	title
1	Davolio	Nancy	Sales Representative
2	Fuller	Andrew	Vice President, Sales
3	Leverling	Janet	Public Relations
4	Peacock	Margaret	Administrative Assistant
5	Buchanan	Steven	Marketing Manager

1 - 5 [Next >>](#)

Saved

To save the created commands, just inform the name of the query and click on the save button. The commands that are already, will be listed.

Saved statements:

 isql_test.sql

Save statement as: [Save](#)

This is a tool that allows the management of your database into ScriptCase. This tool is available to connections with: MySQL, SQLServer(Cannot be ODBC), SQLite, Oracle and PostgreSQL.

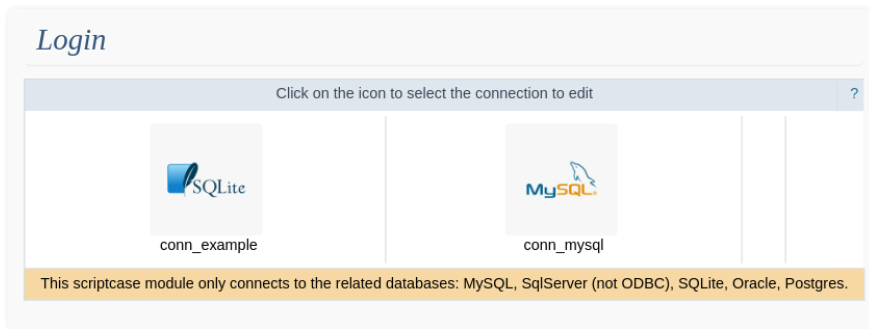
In the beginning we must select an existing connection in the project.

This tool can be accessed in the menu **Database**.

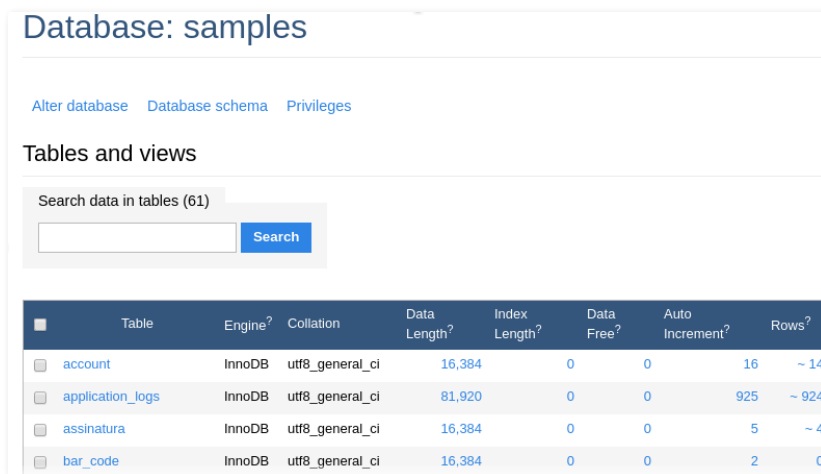
The options and settings on the Database Builder depends of the user database privileges and the used database.

Accessing the Database Builder

First, select the database connection that we wish to display.



Next step, we will go to the home screen from Database Builder, where we can access all the available functionalities.



At the screen left corner, we will see a few functionalities and the table list, to a quickly access of the selected database.

DB

Allows see the database that is being accessed in the moment. Here we can change the database.

To the databases which possess user access restriction, it is necessary that the user set in the connection creation has access to all the databases

SQL Commands

Allows to execute any SQL command, from a select command to procedure commands of creation and execution, for example.

In this case, we will execute a simple SELECT.

MySQL » 192.168.254.84:3357 » samples » SQL command

SQL command

```
SELECT * FROM account
```

[Execute](#) Limit rows: Stop on error Show only errors

Below the code insertion area, we can see a history of commands that were executed.

History

[Edit](#) 13:59:32: SELECT * FROM account;

[Clear](#) [Edit all](#)

If everything goes well, when runs the command, will be showed the consult result, with the [edição dos registros](#) possibility.

```
SELECT * FROM account
```

accountid	accountdescription
1	OFFICE EQUIPMENT
2	FIXED TELEPHONY
3	MOBILE TELEPHONE
4	SOFTWARE
5	SERVICE ADVISOR
6	MAIL
7	CROSSING AIR
8	TAXI
9	TIKET MEAL
10	ELECTRICAL ENERGY
11	WATER
12	RENTAL
13	LODGING
14	HARDWARE

-1 rows (0.005 s) [Edit](#), [Explain](#), [Export](#)

Below of the displayed result, we will see a summary of the used tables in the SQL, informing the number of lines, the quantity of id fields and other informations, even the executed SQL.

17 rows (0.001 s) [Edit](#), [Explain](#), [Export](#)

id?	select_type?	table?	partitions?	type?
1	SIMPLE	account	NULL	ALL

```
SELECT * FROM account
```

To return to the database builder home screen, select the table in the 'breadcrumb' in the page top.

Import Files

Allows you to execute .SQL files.

Import

File upload **From server**

SQL (< 2MB): No file chosen [Execute](#)

Webserver file adminer.sql[.gz] [Run file](#)

Stop on error Show only errors

Export Tables

Enables the creation of a dump in the database.

Export: samples

Output: open save gzip

Format: SQL CSV CSV; TSV

Database: Routines Events

Tables: Auto Increment Triggers

Data: INSERT

Export

<input checked="" type="checkbox"/> Tables	Data <input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> account	~ 17 <input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> application_logs	~ 114 <input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> bar_code	0 <input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> categories	~ 8 <input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> city	~ 2,207 <input checked="" type="checkbox"/>

We must review the default settings of the exportation, for that we can change them and format according with our necessities.

- output - Actions around the exportation.
 - open - Display all the generated SQL inside the ScriptCase.
 - save - Generates a `.sql` file.
 - zip - Generates a `.sql.gz` file.
- format - Format of the created file.
- database - Defines the commands utilization referring at the database creation.
- tables - Defines the commands utilization referring at the database creation.
- data - Defines the commands utilization at the data insertion.

Create Tables

Enables the tables creation through of a simple and intuitive interface, where we can create them quickly.

MySQL » 192.168.254.84:3357 » samples » Create table

Create table

Table name: (engine) (collation) **Save**

Column name	Type	Length	Options	NULL	AI?	+
<input type="text" value="id"/>	int	<input type="text"/>		<input type="checkbox"/>	<input checked="" type="radio"/>	+ ↑ ↓ ×
<input type="text" value="name"/>	varchar	255	(collation)	<input type="checkbox"/>	<input type="radio"/>	+ ↑ ↓ ×
<input type="text"/>	int	<input type="text"/>		<input type="checkbox"/>	<input type="radio"/>	+ ↑ ↓ ×

Auto Increment: Default values Comment

Save

Database Change

Enables change the database name, your default collation or delete the database.

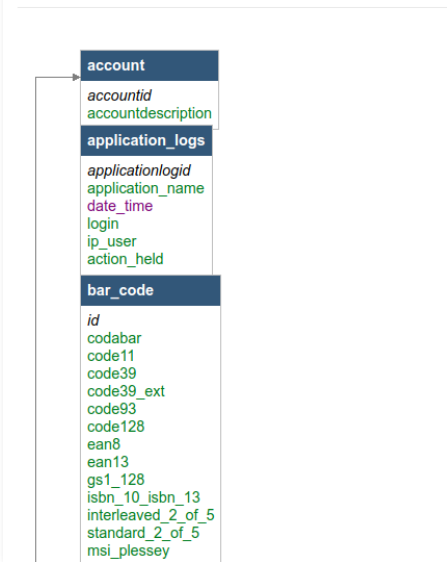
Alter database: samples

utf8_general_ci **Save** **Drop**

Database Schema

Display a relationship diagram of the database tables.

Database schema: samples



Privileges

Edit the users privileges to the current database.

Server: localhost

Username: webhelp

Password: Hashed

Privileges? 'samples'.*

All privileges

Grant option

Server Create user

Server Event

Server Process

Server Proxy

Server Reload

Server Replication client

Server Replication slave

Server Show databases

Server Shutdown

Create View

Enables a quickly creation of a view, where we must inform only the name and the select that will be used in the view.

Create view

Name: view_webhelp

```
SELECT
accountid,
accountdescription
FROM
account
```

Save

After save, we can see the command executed and the structure of the created view.

View: view_webhelp

View has been created. 15:06:48 [SQL command](#)

[Select data](#) [Show structure](#) [Alter view](#) [New item](#)

Column	Type	Comment
accountid	int(11) [0]	
accountdescription	varchar(100) NULL	

All the created views can be displayed together the database tables.

<input type="checkbox"/>	tasks	MyISAM	utf8_general_ci
<input type="checkbox"/>	territories	InnoDB	utf8_general_ci
<input type="checkbox"/>	view_webhelp		View

Create Procedure

This functionality makes the procedure creation more easily. The parameters must be informed like in the example below, where we obligatory must inform the parameter type (input and/or output), your name, the data type that will receive or return and your size.

Name: [Save](#)

	Parameter name	Type	Length
IN	<input type="text" value="id"/>	int	<input type="text"/>
IN	<input type="text"/>	tinyint	<input type="text"/>

```
SELECT * FROM view_webhelp LIMIT id
```

[Save](#)

The executed command will be displayed.

MySQL » 192.168.254.84:3357 » Database: samples

Database: samples

Routine has been created. 15:19:30 [SQL command](#)

[Alter database](#) [Database schema](#) [Privileges](#)

All the created procedures can be visualized below the table and view list.

Routines

Name	Type	Return type
procedure_webhelp	PROCEDURE	Alter

[Create procedure](#) [Create function](#)

Executing A Procedure

There are two ways to execute a procedure inside the database builder.

At visualize the procedure and functions list, we can click in their name. We will be redirect to the procedure call, where we must inform the input parameter(in case that procedure possess).

MySQL » 192.168.254.84:3357 » samples » Call: procedure_webhelp

Call: procedure_webhelp

id

After that, will be displayed the result of the procedure execution and the used SQL for that.

CALL `procedure_webhelp` ('3')

(0.001 s) [Edit](#)

accountid	accountdescription
1	OFFICE EQUIPMENT
2	FIXED TELEPHONY
3	MOBILE TELEPHONE

Routine has been called, -1 rows affected.

id

The second way is using the [SQL command](#) functionality that allow us to execute any SQL command.

In this case, we must use the command to call a procedure: 'CALL procedure_name(parameter)'.

MySQL » 192.168.254.84:3357 » samples » SQL command

SQL command

CALL `procedure_webhelp` ('3')

Limit rows: Stop on error Show only errors

[History](#)

Creating Functions

The parameters must be informed like in the example below, where we obligatory must inform your name, the data type that will receive and the size.

We must define the datatype of the function return too.

Create function

Name:

Parameter name	Type	Length	Options	
<input type="text" value="text"/>	<input type="text" value="varchar"/>	<input type="text" value="50"/>	<input type="text" value="(collation)"/>	<input type="button" value="+"/> <input type="button" value="↑"/> <input type="button" value="↓"/> <input type="button" value="x"/>
Return type	<input type="text" value="varchar"/>	<input type="text" value="50"/>	<input type="text" value="(collation)"/>	

The executed command will be displayed.

MySQL » 192.168.254.84:3357 » Database: samples

Database: samples

Routine has been created. 15:19:30 [SQL command](#)

Routine has been created. 15:20:11 [SQL command](#)

[Alter database](#) [Database schema](#) [Privileges](#)

Executing A Function

There are two ways to execute a function inside the database builder.

At visualize the procedure and functions list, we can click in their name. We will be redirect to the function call, where we must inform the input parameters(in case that possess).

MySQL » 192.168.254.84:3357 » samples » Call: webhelp_function

Call: webhelp_function

text

[Call](#)

After that, will be displayed the result of the execution and the used SQL for that.

MySQL » 192.168.254.84:3357 » samples » Call: webhelp_function

Call: webhelp_function

`SELECT `webhelp_function` ('webhelp')`

(0.001 s) [Edit](#)

``webhelp_function`('webhelp')`
you said: webhelp!!!

text

[Call](#)

The second way is using the [SQL command](#) functionality that allow us to execute any SQL command.

In this case, we must use the command to call a function: 'CALL function_name(parameter)'.

MySQL » 192.168.254.84:3357 » samples » SQL command

SQL command

`SELECT `webhelp_function` ('webhelp')`

[Execute](#) Limit rows: Stop on error Show only errors

API settings let you integrate the various APIs available into the Scriptcase. Through this interface it is possible to configure email sending, SMS and Online Payment APIs.

APIs List

On access you can view a list of all the APIs already configured.

#	Name	Gateway	Actions
1	yahoo	smtp	Edit Delete
2	gmail	smtp	Edit Delete

[Add new](#)

- **ame**: Configured API Name.
- **etaway**: Getaway used to send the email.

Email APIs

Scriptcase connects to three different APIs for sending emails: SMTP, Mandrill, and Amazon SES.

SMTP Configuration

SMTP is the standard protocol for sending e-mails over the Internet, and each provider has its SMTP.

For more information about SMTP settings [click here](#)

Name	<input type="text" value="Name"/>
Gateway	<input type="text" value="smtp"/>
SMTP server	<input type="text" value="smtp.example.com"/>
SMTP port	<input type="text" value="465"/>
SMTP user	<input type="text" value="default@example.com"/>
SMTP password	<input type="password" value="*****"/>
SMTP protocol	<input type="text" value="SSL"/>
From email	<input type="text" value="default@example.com"/>
From name	<input type="text" value="default"/>
<input type="button" value="Close"/> <input type="button" value="Save changes"/>	

API

It allows you to select an API already configured in **Tools > API** or set a new one here by selecting **- custom -**. In this case, you see some options according to the selected Gateway.

Gateway

Select the way to send the email. The SMTP is the default protocol for sending e-mails over the Internet, and each provider has its SMTP.

SMTP Server

SMTP server address for the provider.

Port SMTP

Port used by the mail server. Use port 465 for security with SSL, port 587 for security with TLS, or port 25 as port without security. By omitting the value, Scriptcase defaults to 25.

User SMTP

SMTP server user.

Enter SMTP

SMTP server user password.

Protocol SMTP

Defines the security protocol. By omitting the value, Scriptcase uses the default value.

E-mail

Enter the origin email, that is the email sender.

Name

The sender's name displayed in the email.

Mandrill Configurations

Mandrill is a transactional email API for MailChimp users, ideal for sending data-driven emails.

Name	<input type="text" value="Name"/>
Gateway	<input type="text" value="mandrill"/>
API key	<input type="text" value="Your API"/>
From email	<input type="text" value="default@example.com"/>
From name	<input type="text" value="default"/>
<input type="button" value="Close"/> <input type="button" value="Save changes"/>	

API

It allows you to select an API already configured in **Tools > API** or set a new one here by selecting **- custom -**. In this case, you see some options according to the selected Gateway.

Gateway

Select the way to send the email. **Mandrill** is a transactional email API for MailChimp users, ideal for sending data-driven emails.

API KEY

Enter the key obtained from the origin site of your API.

E-mail

Enter the origin email, that is the email sender.

Name

The sender's name displayed in the email.

For more information about [mandrill](#)

Amazon SES Configurations

Amazon Simple Email Service **Amazon SES** is an email sending service designed to assist in sending marketing emails, notifications, and transactional messages.

Name	<input type="text" value="Name"/>
Gateway	<input type="text" value="Amazon SES"/>
API key	<input type="text" value="Your Key API"/>
API Secret	<input type="text" value="Your Secret API"/>
Region	<input type="text" value="Region"/>
From email	<input type="text" value="default@example.com"/>
From name	<input type="text" value="default"/>
<input type="button" value="Close"/> <input type="button" value="Save changes"/>	

API KEY

Enter the key obtained from the origin site of your API.

API Secret

Enter the secret access key of your account.

Region

Amazon SES has endpoints in many countries, to reduce network latency, inform the region of the endpoint closest to your application. [See the regions.](#)

E-mail

Enter the origin email, that is the email sender.

Name

The sender's name displayed in the email.

For more information about [Amazon SES](#)

SMS APIs

Scriptcase connects to different APIs for sending SMS text messages:

- [Twilio](#)
- [Plivo](#)

- [Clickatell](#)

Setting Up Twilio API

Twilio is a set of libraries that enables SMS communication on a global scale. For correct configuration, we need to enter the data of **Authentication ID**, **Authentication Token**, and **Sender**, as shown below:

Name	<input type="text" value="Name"/>
Gateway	<input type="text" value="twilio"/>
Auth ID	<input type="text" value="Your Auth ID"/>
Auth Token	<input type="text" value="Your Auth Token"/>
From	<input type="text" value="From number"/>
<input type="button" value="Close"/> <input type="button" value="Save changes"/>	

Name

Enter a name for your API;

Gateway

The Gateway we want to use. (in this case, Twilio);

Auth ID

Enter the authentication ID for your Twilio account;

Auth Token

Enter the authentication Token for your Twilio account;

From

Enter the sender's number to send the SMS;

Setting Up Plivo API

Name	<input type="text" value="Name"/>
Gateway	<input type="text" value="plivo"/>
Auth ID	<input type="text" value="Your Auth ID"/>
Auth Token	<input type="text" value="Your Auth Token"/>
From	<input type="text" value="From number"/>
<input type="button" value="Close"/> <input type="button" value="Save changes"/>	

Name

Enter a name for your API;

Gateway

The Gateway we want to use. (in this case, Plivo);

Auth ID

Enter the authentication ID of your Plivo account;

Token ID

Enter the authentication Token of your Plivo account;

From

Enter the sender number to send the SMS;

Setting Up Clickatell API

Name	<input type="text" value="Name"/>
Gateway	<input type="text" value="clickatell"/>
Auth Token	<input type="text" value="Your Auth Token"/>
<input type="button" value="Close"/> <input type="button" value="Save changes"/>	

Name

Enter a name for your API;

Gateway

The Gateway we want to use. (in this case, Clickatell);

Auth Token

Enter the authentication Token of your Clickatell account;

Watch bellow a video with an example for SMS notifications:

Whatsapp APIs

We use the [ChatAPI](#) for sending messages through WhatsApp.

Chat API

First, we should [\[access the ChatAPI website\]\[get_token_chatapi\]{}{target='blank'}](#) to get the data information for the API configuration in Scriptcase.

With the data in hand, go to the **Tools> API** menu to access the settings screen (as shown below in the image).

Name	api_waboxapp
Mode	Public
Gateway	chatapi
URL	[blurred]
Auth Token	[blurred]

Name

Enter a name to identify the configured API. We recommend to use a name that identifies the data of configuration since it is possible to create more than one API with this Gateway.

Mode

Defines the API access level: **Public**, where all users of all projects have access to this API; **Project**, where all users of this project have access to this API; **User**, where only the user creating the API has access to this API.

Gateway

Enter the Gateway to use. We must select Chat api.

- **RL** - URL required for sending a message

Auth Token

Enter the Token obtained from the settings of the API origin site. [\[Visit the ChatAPI website to get your token.\]\[get_token_chatapi\]{}{target='blank'}](#)

After configuring the API, we must use the macro [macro sc_whatsapp \(\)](#); to complete the information to send the message.

Check the macro syntax below:

```
sc_send_whatsapp ([ 'profile' => 'api_chatapi',
                    'message' => 'Message to be sent',
                    'to' => '558182499699' ] );
```

Watch a video showing an example with ChatAPI:

Payment APIs

Scriptcase allows using APIs for online payment processing. The following API integrations are available:

- [paypal](#)

Setting Up PayPal

Name	<input type="text" value="Name"/>
Gateway	<input type="text" value="paypal_express"/>
Username	<input type="text" value="Username"/>
Password	<input type="text" value="Password"/>
Signature	<input type="text" value="Signature"/>
Test Mode	<input type="text" value="FALSE"/>

Name

Enter a name for your API;

Gateway

Enter the Gateway to use (For example, paypal_express);

Username

Enter the user of PayPal account for authentication;

Password

Enter the password of PayPal account for authentication;

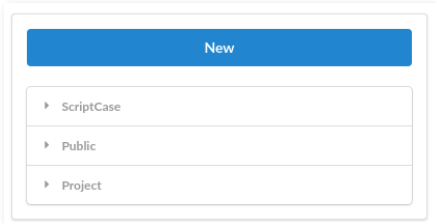
Signature

Enter the signature of PayPal account for authentication;

Test Mode

It allows you to use the API as a test mode.

This tool allows that the user manage templates to the application hotkeys. Can be possible create a new hotkeys set or use an existing setting and change then to improve your development environment.



Home screen to the template and scopes managements available.

Available Scopes:

- **criptcase:** Here is the default Scriptcase template. This template cannot be edited or erased, only can visualized or a copy can be made to utilize the already defined hotkeys.
- **ublic:** Here is the created and edited templates at public level by user.
- **roject:** Here is the created and edited templates at project level by user.

In "New" the user will can create a new template with all the configurable hotkeys.

Then, will come the creation screen, when the user will can select the name and the scope of the template that will be created.

Creation screen of Hotkey Templates

- **ode:** Allows the user to select the scope that the template will be created.
- **ame:** Allows the user to select the name to the template that will be created.
- **ction:** Selects the action that will be made when press the selected keybind.
- **eybinding:** Selects the keybinds that will be responsible for execute the chosen action.
- **dd "+":** Adds a new action in the keybind list.
- **lear:** Clears all predefined settings by Scriptcase or by user.

When selecting one of the categories that possess created templates, the user will have some setting options. All the applications will come with the Scriptcase default template configured.



Options to Template configuration.

- **dit:** Allows the user to edit the name, mode and the defined hotkeys in the template.
- **opy:** Creates a copy of the selected template allowing to use the same configurations, change the name and hotkeys settings that the user desires.
- **elate:** Realizes the exclusion of the selected template.

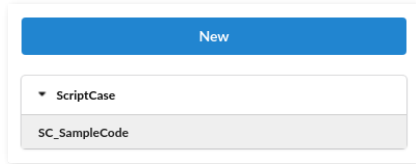
When editing the name or the scope of a template, automatically all the related applications to them will lose your own configurations. Will be necessary configure the edited template again as in use in the application.

This tool allows the developer to perform the creation of blocks of code and also to use these codes easily in Scriptcase applications. In this way, streamlining and facilitating the use of programming within the tool.

The created codes can be used in all events present in Scriptcase applications. Through the **Snippets** tab.

Creating a Code Snippet

To create a Code Snippet, simply click the **New** button available in the interface.



View:

It allows the user to view all the code, but without the possibility to make edits.

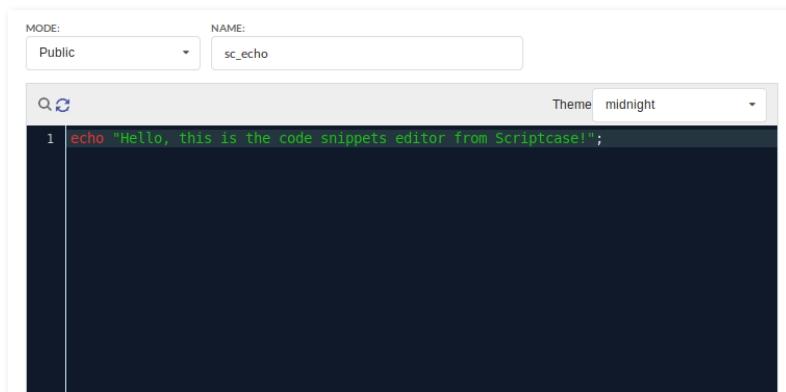
Copy:

It makes a copy of the code snippet for a new one, allowing the user to change where it will be located and the usage name.

Edit:

It allows the complete editing of the selected code, as well as where it is located and its name of use. This option is only available for the code snippets located in Public, Project, and User.

After that, the code edit screen will open, where you can create the code desired by the developer. See the image below:



Mode:

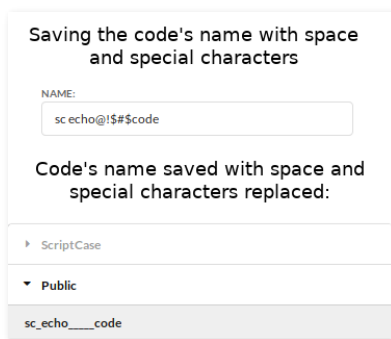
Lets you select where the code will be created. As available options there are: Public, Project and User.

Name:

Lets you define the name that will be used to call the code in the event.

Any special character and/or space that is entered in the name is automatically replaced by the symbol (_).

Example:



Themes:

Allows you to select the theme used in the code editor. If you already have a previously selected theme for editing an event, the same theme is applied to the code snippet editor.

Search:

The search icon  allows the user to search for one or more terms within the code.

Replace:

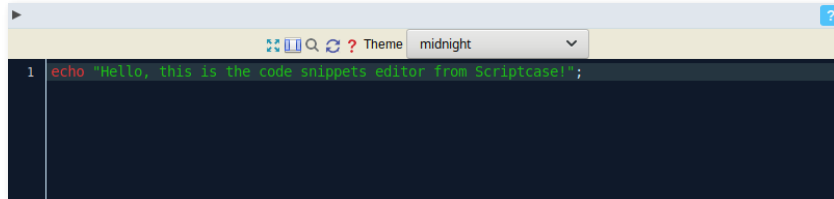
The replace icon  allows you to replace snippets of code with another. Thus, facilitating the editing of several terms within the code.

Using a Code Snippet

To use a code snippet you must have access to any event present in the Scriptcase. By accessing the **Snippets** tab, you can view the list of code snippets created by Scriptcase and the user.



When you click on the usage name, the code is automatically added to the event.



To proceed with the password recovery, it is necessary that has the SMTP server configured for a successful process. Look [here](#) how to realize this procedure.

Yahoo

For more information about these settings [Click Here](#)

SMTP Server Settings Yahoo	
SMTP Server	smtp.mail.yahoo.com
Port	465 or 587
E-mail address	Your full email address (name@domain.com.)
Password	The password for your account.
Requires SSL	Yes
Requires TSL	Yes (if available)
Requires authentication	Yes

Some email applications uses older, less secure technologies to sign in to your email account, and Google will block these requests by default. To solve this, sign in to your Google account and enable access through less secure apps through this [link](#).

Gmail

For more information about these settings [Click Here](#)

SMTP Server Settings Gmail	
SMTP Server	smtp.gmail.com
E-mail address	Your full email address (Ex. example@gmail.com)
Password	The password for your account.
Port (TLS)	587
Port (SSL)	465
TLS / SSL Required	Yes

Some email applications uses older, less secure technologies to sign in to your email account, and Google will block these requests by default. To solve this, sign in to your Google account and enable access through less secure apps through this [link](#).

Outlook

For more information about these settings [Click Here](#)

SMTP Server Settings Outlook.com	
SMTP Server	smtp-mail.outlook.com
E-mail address	Your full email address (Ex. myname@outlook.com, not an alias)
Password	The password for your account.
Port	587 (you can use port 25 as an alternative)
TLS / SSL Required	Yes

Hotmail

For more information about these settings [Click Here](#)

SMTP Server Settings Windows Live Hotmails	
SMTP Server	smtp.live.com
E-mail address	Your full email address (Ex. me@hotmail.com or me@live.com)
Password	The password for your account.
Port	587
TLS / SSL Required	Yes

In Layout section, you can either edit or create standard previews that will be placed in the developed project with ScriptCase. In this area, you can make changes to each part of the application, being from presentation of the applications to the buttons and menu on which they'll be presented in.

In the CSS Applications section, it's possible to create layouts for the applications using the editor. It's possible to create new themes or edit existing ones.

Theme gallery Create new theme ? **Sc8_BlueWood**

ScriptCase	
Sc8_BlueWood	View Save as
Sc5_Red	View Save as
Sc5_Green	View Save as
Sc5_Lilac	View Save as
Sc5_Blue	View Save as
Sc5_Silver	View Save as
Sc4_SoftOrange	View Save as
Sc4_Plain	View Save as
Sc4_Olive	View Save as
Sc4_Musgo	View Save as
Sc4_Money	View Save as
Sc4_Sunny	View Save as
Sc4_Sunny	View Save as
Sc4_SoftOrange	View Save as
Sc4_Plain	View Save as
Sc4_Olive	View Save as
Sc4_Musgo	View Save as
Sc4_Money	View Save as
Sc4_LighGray	View Save as
Sc4_LighBrown	View Save as
Sc4_LighBlueOcean	View Save as
Sc4_LighBlue	View Save as
Sc4_Cherry	View Save as

Header

First Previous Next Last Add Save

Error field xxx xxxx xxxxxxxxxxxx

Block 1.1

Title 1

Title 2

Block 2.1

Title 1	Title 2	Title 3
11111111	22222222	33333333
11111111	22222222	33333333
11111111	22222222	33333333
11111111	22222222	33333333

Footer

To create new themes, click on "Create New Theme". And the modifications done to the theme, will be displayed on the preview in real-time.

Theme editor Save Cancel **asd**

Theme settings

Page

Button

Menu Icon Schema

Background color

Font

Corner

Content

Header

Toolbar

Table

Block

Object

Label

Header

First Previous Next Last Add Save

Error field xxx xxxx xxxxxxxxxxxx

Block 1.1

Title 1

Title 2

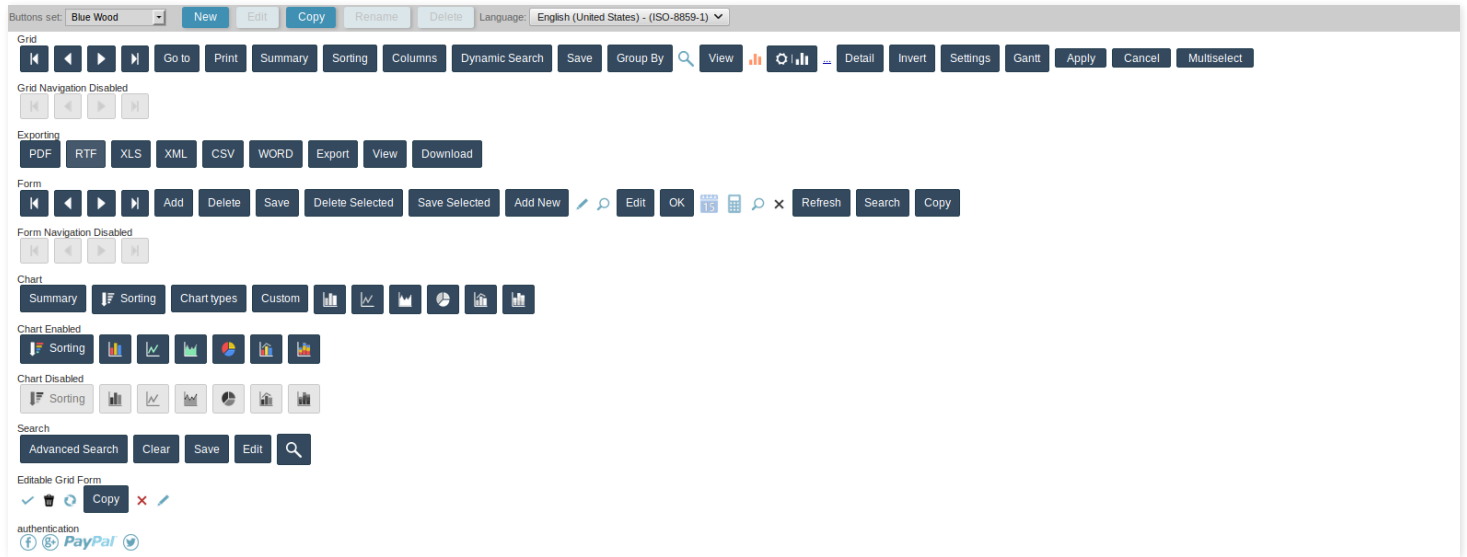
Block 2.1

Title 1	Title 2	Title 3
11111111	22222222	33333333
11111111	22222222	33333333
11111111	22222222	33333333
11111111	22222222	33333333

Footer

General Overview

In Button Editing we can create new sets of buttons to associate them with themes. We can create or edit sets of text and image buttons.



Creating a New Button Scheme

Click the "New" button on the Button Scheme toolbar.

Preview	Type	Display	Image position	Label	Hint	CSS Style
Grid						
First page	Button	Only text		{lang_btms_frst}	{lang_btms_frst_hint}	default
Previous page	Button	Only text		{lang_btms_prev}	{lang_btms_prev_hint}	default
Next page	Button	Only text		{lang_btms_next}	{lang_btms_next_hint}	default
Last page	Button	Only text		{lang_btms_last}	{lang_btms_last_hint}	default
Go to line	Button	Only text		{lang_btms_jump}	{lang_btms_jump_hint}	default
Printable Version	Button	Only text		{lang_btms_prnt}	{lang_btms_prnt_hint}	default
Display Summary	Button	Only text		{lang_btms_smry}	{lang_btms_smry_hint}	default
Sorting	Button	Only text		{lang_btms_sort}	{lang_btms_sort_hint}	default
Column	Button	Only text		{lang_btms_clmn}	{lang_btms_clmn_hint}	default
Dynamic Search	Button	Only text		{lang_btms_dynamicsearch}	{lang_btms_dynamicsearch}	default

In the first tab (Buttons) the buttons used in Scriptcase are displayed. Each button can have a different characteristic, being possible to choose between image, text button or link. If you use the image type button, a field will be displayed to upload the image. For text and link button types use the tabs: Button Style and Link Styles to edit their appearances.

Editing button styles

Buttons Button Styles Link Styles

CSS Button Settings

CSS Style: default New

Font settings Font: <input type="text" value="Tahoma, Arial, sans-serif"/> Font Size: <input type="text" value="11px"/> Font color: <input type="text" value="#000000"/> Font Weight: <input type="text" value="normal"/> Line height: <input type="text"/>	Border settings Border color: <input type="text" value="#dcdcdc"/> Border Width: <input type="text" value="1px"/> Border Style: <input type="text" value="solid"/> Border radius: <input type="text" value="4px"/>	Preview
Background settings Background Color: <input type="text" value="#EEEEEE"/> Background Image: <input type="text"/>	Others Height: <input type="text"/> Padding: <input type="text" value="4px 8px"/> Text shadow: <input type="text"/> Text-decoration: <input type="text" value="none"/> Cursor: <input type="text"/>	

In this tab you can define the CSS style of the text buttons used in your outline. All schemes already have a default style that can not be deleted. If your layout requires more than one type of layout for the text buttons, just click on the "new" button and set up a new style. Go back to the Buttons tab and select the created style.

Configuring Link Button Style

Buttons Button Styles Link Styles

CSS Link Settings

CSS Style: default New

Link Font: <input type="text"/> Font Size: <input type="text" value="12px"/> Font color: <input type="text" value="#0000AA"/> Text-decoration: <input type="text" value="underline"/> Cursor: <input type="text"/> Preview: Link Example	Visited Font: <input type="text"/> Font Size: <input type="text" value="12px"/> Font color: <input type="text" value="#0000AA"/> Text-decoration: <input type="text" value="underline"/> Cursor: <input type="text"/> Preview: Link Example
Active Font: <input type="text"/> Font Size: <input type="text" value="12px"/> Font color: <input type="text" value="#0000AA"/> Text-decoration: <input type="text" value="underline"/> Cursor: <input type="text"/> Preview: Link Example	Hover Font: <input type="text"/> Font Size: <input type="text" value="12px"/> Font color: <input type="text" value="#0000AA"/> Text-decoration: <input type="text" value="none"/> Cursor: <input type="text"/> Preview: Link Example

Using the same principle exposed in the item styles of text buttons is applied to the styles of links. So you can define that one or more buttons will be of type link and define your CSS style or a new one that you will create.



1. [Home](#)
2. Layout

General View

Creating new buttons set, make possible to edit each component from the Buttons theme so the user can change directly all the sets of buttons, links, and their styles.



Buttons

Here is allowed to configure the default buttons to the applications and components.

See below the available items:

[Grid](#) [Authentication](#)
[Export Email](#) [Menu](#)
[Exporting](#) [Others](#)
[Form](#) [Panels And Modal](#)
[Chart](#) [Calendar](#)
[Chart](#)
[Enabled](#) [SweetAlert](#)
[Search](#) [Editable Grid Form](#)

Button Styles

In this option it is possible to edit the styles that will be used in the buttons set, being available to the user with the following settings:

Button

[Button](#) [Button](#)
 [Disabled](#)
[Button Hover](#) [Button Selected](#)
[Button](#)
[Onclick](#)

Dropdown

[Container](#)
[Line](#)
[Button Hover](#)

Link Styles

In this option, it is possible to edit the styles that will be used in the buttons links.

[Link Styles](#)

RTL Flag

This option allows that be defined the writing orientation used by the theme. Can be defined if the option **RTL(Right to Left)** will be used.





1. [Home](#)
2. [Layout](#)
3. Grid

Grid

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Grid Navigation Disabled

Grid Navigation Disabled

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Export Email

Export Email

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

□

1. [Home](#)
2. [Layout](#)
3. Exporting

Exporting

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

□

1. [Home](#)
2. [Layout](#)
3. Form

Form

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Form Navigations Disabled

Form Navigations Disabled

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. [Chart](#)

Chart

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

□

1. [Home](#)
2. [Layout](#)
3. Chart Enabled

Chart Enabled

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Chart Disabled

Chart Disabled

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Search

Search

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

□

1. [Home](#)
2. [Layout](#)
3. Editable Grid Form

Editable Grid Form

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Authentication

Authentication

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Menu

Menu

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Others

Others

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Modal/App Div

Modal/App Div

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. [Calendar](#)

Calendar

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. SweetAlert

SweetAlert

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

1. [Home](#)
2. [Layout](#)
3. Button

Button

Font

Family

This option allows you to select the family and the font type that will be used in the button.

Color

This option allows you to select the color that will be shown in the button.

Size

This option allows you to select the button font-size in pixels.

Weight

This option allows you to configure the button **font-weight**.

Shadow

This option allows you to configure the shadow of the button font.

Decoration

This option allows you to select an underline type for the button font.

Style

This option allows you to select the style type that will be applied to the button font.

Border

Width

This option allows you to define the button border width size.

Color

This option allows you to define the button border color.

Style

This option allows you to select the style type that will be applied to the button border.

Radius

This option allows you to configure the border button radius.

Background

Color

This option allows you to define the button background color.

Image

This option allows you to define a image to the button background.

Repeat

This option allows you to configure the repeating for the image selected to the button.

Position

This option allows you to configure the selected image align for the button.

Shadow

This option allows you to configure the shadow of the button background.

Others

Opacity

This option allows you to configure the button opacity.

Height

This option allows you to configure the button height.

Padding

This option allows you to configure the button padding element.

Margin

This option allows you to configure the button margin.

Cursor

This option allows you to select the cursor type that will be shown when you pass the mouse over the button.

Transition

This option allows you to change the button values with a specific duration.



1. [Home](#)
2. [Layout](#)
3. Button Hover

Button Hover

Font

Family

This option allows you to select the family and the font type that will be used in the button.

Color

This option allows you to select the color that will be shown in the button.

Size

This option allows you to select the button font-size in pixels.

Weight

This option allows you to configure the button **font-weight**.

Shadow

This option allows you to configure the shadow of the button font.

Decoration

This option allows you to select an underline type for the button font.

Style

This option allows you to select the style type that will be applied to the button font.

Border

Width

This option allows you to define the button border width size.

Color

This option allows you to define the button border color.

Style

This option allows you to select the style type that will be applied to the button border.

Radius

This option allows you to configure the border button radius.

Background

Color

This option allows you to define the button background color.

Image

This option allows you to define a image to the button background.

Repeat

This option allows you to configure the repeating for the image selected to the button.

Position

This option allows you to configure the selected image align for the button.

Shadow

This option allows you to configure the shadow of the button background.

Others

Opacity

This option allows you to configure the button opacity.

Height

This option allows you to configure the button height.

Padding

This option allows you to configure the button padding element.

Margin

This option allows you to configure the button margin.

Cursor

This option allows you to select the cursor type that will be shown when you pass the mouse over the button.

Transition

This option allows you to change the button values with a specific duration.



1. [Home](#)
2. [Layout](#)
3. Button Onclick

Button Onclick

Font

Family

This option allows you to select the family and the font type that will be used in the button.

Color

This option allows you to select the color that will be shown in the button.

Size

This option allows you to select the button font-size in pixels.

Weight

This option allows you to configure the button **font-weight**.

Shadow

This option allows you to configure the shadow of the button font.

Decoration

This option allows you to select an underline type for the button font.

Style

This option allows you to select the style type that will be applied to the button font.

Border

Width

This option allows you to define the button border width size.

Color

This option allows you to define the button border color.

Style

This option allows you to select the style type that will be applied to the button border.

Radius

This option allows you to configure the border button radius.

Background

Color

This option allows you to define the button background color.

Image

This option allows you to define a image to the button background.

Repeat

This option allows you to configure the repeating for the image selected to the button.

Position

This option allows you to configure the selected image align for the button.

Shadow

This option allows you to configure the shadow of the button background.

Others

Opacity

This option allows you to configure the button opacity.

Height

This option allows you to configure the button height.

Padding

This option allows you to configure the button padding element.

Margin

This option allows you to configure the button margin.

Cursor

This option allows you to select the cursor type that will be shown when you pass the mouse over the button.

Transition

This option allows you to change the button values with a specific duration.



1. [Home](#)
2. [Layout](#)
3. Button Disabled

Button Disabled

Font

Family

This option allows you to select the family and the font type that will be used in the button.

Color

This option allows you to select the color that will be shown in the button.

Size

This option allows you to select the button font-size in pixels.

Weight

This option allows you to configure the button **font-weight**.

Shadow

This option allows you to configure the shadow of the button font.

Decoration

This option allows you to select an underline type for the button font.

Style

This option allows you to select the style type that will be applied to the button font.

Border

Width

This option allows you to define the button border width size.

Color

This option allows you to define the button border color.

Style

This option allows you to select the style type that will be applied to the button border.

Radius

This option allows you to configure the border button radius.

Background

Color

This option allows you to define the button background color.

Image

This option allows you to define a image to the button background.

Repeat

This option allows you to configure the repeating for the image selected to the button.

Position

This option allows you to configure the selected image align for the button.

Shadow

This option allows you to configure the shadow of the button background.

Others

Opacity

This option allows you to configure the button opacity.

Height

This option allows you to configure the button height.

Padding

This option allows you to configure the button padding element.

Margin

This option allows you to configure the button margin.

Cursor

This option allows you to select the cursor type that will be shown when you pass the mouse over the button.

Transition

This option allows you to change the button values with a specific duration.

1. [Home](#)
2. [Layout](#)
3. Button Selected

Button Selected

Font

Family

This option allows you to select the family and the font type that will be used in the button.

Color

This option allows you to select the color that will be shown in the button.

Size

This option allows you to select the button font-size in pixels.

Weight

This option allows you to configure the button **font-weight**.

Shadow

This option allows you to configure the shadow of the button font.

Decoration

This option allows you to select an underline type for the button font.

Style

This option allows you to select the style type that will be applied to the button font.

Border

Width

This option allows you to define the button border width size.

Color

This option allows you to define the button border color.

Style

This option allows you to select the style type that will be applied to the button border.

Radius

This option allows you to configure the border button radius.

Background

Color

This option allows you to define the button background color.

Image

This option allows you to define a image to the button background.

Repeat

This option allows you to configure the repeating for the image selected to the button.

Position

This option allows you to configure the selected image align for the button.

Shadow

This option allows you to configure the shadow of the button background.

Others

Opacity

This option allows you to configure the button opacity.

Height

This option allows you to configure the button height.

Padding

This option allows you to configure the button padding element.

Margin

This option allows you to configure the button margin.

Cursor

This option allows you to select the cursor type that will be shown when you pass the mouse over the button.

Transition

This option allows you to change the button values with a specific duration.



1. [Home](#)
2. [Layout](#)
3. Container

Container

Into the Scriptcase buttons settings we can find the main layout settings among them we can find the 'Dropdown' and 'Button' options.

If what we need is to customize the dropdown style menu then we can select the option 'Dropdown' then some interesting options are available:

- Container
- Line
- Button Hover
- Button Disabled
- Button Selected
- Button OnClick



The container is the border of the menu, we can change some settings including:

- Border
- Background
- Others

Border

The border offer us some settings:



Width

This option allows you to define the button border width size.

Color

This option allows you to define the container border color.



Style

This option allows you to select the style type that will be applied to the button border.

Radius

This option allows you to configure the border button radius.

Background



Color

This option allows you to define the button background color.

Image

This option allows you to define a image to the button background.

Repeat

This option allows you to configure the repeating for the image selected to the button.

Position

This option allows you to configure the selected image align for the button.

Shadow

This option allows you to configure the shadow of the button background.

Others



Opacity

This option allows you to configure the button opacity.

Padding

This option allows you to configure the button padding element.

Margin

This option allows you to configure the button margin.

Cursor

This option allows you to select the cursor type that will be shown when you pass the mouse over the button.



1. [Home](#)
2. [Layout](#)
3. Line

Line

Into the Scriptcase buttons settings we can find the main layout settings among them we can find the 'Dropdown' and 'Button' options.

If what we need is to customize the dropdown style menu then we can select the option 'Dropdown' then some interesting options are available:

- Container
- Line
- Button Hover
- Button Disabled
- Button Selected
- Button OnClick



The container is the border of the menu, we can change some settings including:

- Border
- Background
- Others

Border

The border offer us some settings:



Width

This option allows you to define the button border width size.

Color

This option allows you to define the container border color.



Style

This option allows you to select the style type that will be applied to the button border.

Radius

This option allows you to configure the border button radius.

Background



Color

This option allows you to define the button background color.

Image

This option allows you to define a image to the button background.

Repeat

This option allows you to configure the repeating for the image selected to the button.

Position

This option allows you to configure the selected image align for the button.

Shadow

This option allows you to configure the shadow of the button background.

Others



Opacity

This option allows you to configure the button opacity.

Padding

This option allows you to configure the button padding element.

Margin

This option allows you to configure the button margin.

Cursor

This option allows you to select the cursor type that will be shown when you pass the mouse over the button.



1. [Home](#)
2. [Layout](#)
3. Button Hover

Button Hover

Into the Scriptcase buttons settings we can find the main layout settings among them we can find the 'Dropdown' and 'Button' options.

If what we need is to customize the dropdown style menu then we can select the option 'Dropdown' then some interesting options are available:

- Container
- Line
- Button Hover
- Button Disabled
- Button Selected
- Button OnClick



The container is the border of the menu, we can change some settings including:

- Border
- Background
- Others

Border

The border offer us some settings:



Width

This option allows you to define the button border width size.

Color

This option allows you to define the container border color.



Style

This option allows you to select the style type that will be applied to the button border.

Radius

This option allows you to configure the border button radius.

Background



Color

This option allows you to define the button background color.

Image

This option allows you to define a image to the button background.

Repeat

This option allows you to configure the repeating for the image selected to the button.

Position

This option allows you to configure the selected image align for the button.

Shadow

This option allows you to configure the shadow of the button background.

Others



Opacity

This option allows you to configure the button opacity.

Padding

This option allows you to configure the button padding element.

Margin

This option allows you to configure the button margin.

Cursor

This option allows you to select the cursor type that will be shown when you pass the mouse over the button.



1. [Home](#)
2. [Layout](#)
3. Link Styles

Link Styles

CSS Link Settings

Here we can configure the links that will be shown as Buttons in Scriptcase. Handling various visual aspects related to these links.

Also is allowed to create new CSS Link Styles clicking in the button **New**.

Link

Font

This option allows you to configure the font type that will be used in the link.

Font Size

This option allows you to configure the font size that will be used in the link.

Font Color

This option allows you to configure the link color for the exhibition.

Text-decoration

This option allows you to configure the line personalization type that the link will have.

Cursor

This option allows you to configure the cursor type that will be shown when passing the mouse through the link.

Visited

Font

This option allows you to configure the font type that will be used in the link.

Font Size

This option allows you to configure the font size that will be used in the link.

Font Color

This option allows you to configure the link color for the exhibition.

Text-decoration

This option allows you to configure the line personalization type that the link will have.

Cursor

This option allows you to configure the cursor type that will be shown when passing the mouse through the link.

Active

Font

This option allows you to configure the font type that will be used in the link.

Font Size

This option allows you to configure the font size that will be used in the link.

Font Color

This option allows you to configure the link color for the exhibition.

Text-decoration

This option allows you to configure the line personalization type that the link will have.

Cursor

This option allows you to configure the cursor type that will be shown when passing the mouse through the link.

Hover

Font

This option allows you to configure the font type that will be used in the link.

Font Size

This option allows you to configure the font size that will be used in the link.

Font Color

This option allows you to configure the link color for the exhibition.

Text-decoration

This option allows you to configure the line personalization type that the link will have.

Cursor

This option allows you to configure the cursor type that will be shown when passing the mouse through the link.



1. [Home](#)
2. Layout

General View

Creating new buttons set, make possible to edit each component from the Buttons theme so the user can change directly all the sets of buttons, links, and their styles.

Buttons

Here is allowed to configure the default buttons to the applications and components.

See below the available items:

[Grid](#) [Authentication](#)
[Export Email](#) [Menu](#)
[Exporting](#) [Others](#)
[Form](#) [Panels And Modals](#)
[Chart](#) [Calendar](#)
[Chart](#)
[Enabled](#) [SweetAlert](#)
[Search](#) [Editable Grid Form](#)

Button Styles

In this option it is possible to edit the styles that will be used in the buttons set, being available to the user with the following settings:

Button

[Button](#) [Button](#)
 [Disabled](#)
[Button Hover](#) [Button Selected](#)
[Button](#)
[Onclick](#)

Dropdown

[Container](#)
[Line](#)
[Button Hover](#)

Link Styles

In this option, it is possible to edit the styles that will be used in the buttons links.

[Link Styles](#)

RTL Flag

This option allows that be defined the writing orientation used by the theme. Can be defined if the option **RTL(Right to Left)** will be used.

□

1. [Home](#)
2. [Layout](#)
3. Grid

Grid

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

□

1. [Home](#)
2. [Layout](#)
3. Grid Navigation Disabled

Grid Navigation Disabled

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Export Email

Export Email

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Exporting

Exporting

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Form

Form

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Form Navigations Disabled

Form Navigations Disabled

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

□

1. [Home](#)
2. [Layout](#)
3. [Chart](#)

Chart

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

□

1. [Home](#)
2. [Layout](#)
3. Chart Enabled

Chart Enabled

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Chart Disabled

Chart Disabled

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Search

Search

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

□

1. [Home](#)
2. [Layout](#)
3. Editable Grid Form

Editable Grid Form

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Authentication

Authentication

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

□

1. [Home](#)
2. [Layout](#)
3. Menu

Menu

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Others

Others

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Modal/App Div

Modal/App Div

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. [Calendar](#)

Calendar

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. SweetAlert

SweetAlert

In this option we can configure the layout of the buttons that will be displayed in the applications used in the project.

Below you can see the list of buttons you can configure.

Type

In this option you can define how the button will appear in the application.

The available options are:

[Image](#)

If you select Image, the button appears as an icon in the application.

[Button](#)

If you select the button type you can perform the button configuration that will be displayed in the applications.

[Link](#)

If you select the Link Type the button in the application will be displayed as a link.

Image



Option Preview

It shows a preview of how the button will appear in the application according to the settings made.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Button



Display

Option to set what will be displayed on the button, the options are:

Only Text

Only the text defined by the user will be displayed.

Only Image

Only the image defined by the user will be displayed.

Text and Image

Both text and images defined by the user will be displayed.

Only Font-awesome icon

The image selected by the user will be displayed.

Display text and icon Font-awesome

Both text and images defined by the user will be displayed.

Icon

Option that allows the upload of own icons to the Scriptcase image manager, in order to be used in the buttons.

Image Position

Sets where the image will be placed on the button, the options are:

Image on the right

The image will be displayed on the right and the text will be displayed on the left.

Text on the right

The text will be displayed on the right and the image will be displayed on the left.

Label

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.

Link**Label**

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

Hint

Option to enter text configured in [Translate Applications][translate_applications] or text entered by the user that will be shown when the mouse is over the button.

CSS Style

Option to set which style will be used by buttons. Layouts are created in the Button Styles or Link Styles options.



1. [Home](#)
2. [Layout](#)
3. Button

Button

Font



Family

This option allows you to select the family and the font type that will be used in the button.

Color

This option allows you to select the color that will be shown in the button.

Size

This option allows you to select the button font-size in pixels.

Weight

This option allows you to configure the button **font-weight**.

Shadow

This option allows you to configure the shadow of the button font.

Decoration

This option allows you to select an underline type for the button font.

Style

This option allows you to select the style type that will be applied to the button font.

Border



Width

This option allows you to define the button border width size.

Color

This option allows you to define the button border color.

Style

This option allows you to select the style type that will be applied to the button border.

Radius

This option allows you to configure the border button radius.

Background



Color

This option allows you to define the button background color.

Image

This option allows you to define a image to the button background.

Repeat

This option allows you to configure the repeating for the image selected to the button.

Position

This option allows you to configure the selected image align for the button.

Shadow

This option allows you to configure the shadow of the button background.

Others



Opacity

This option allows you to configure the button opacity.

Height

This option allows you to configure the button height.

Padding

This option allows you to configure the button padding element.

Margin

This option allows you to configure the button margin.

Cursor

This option allows you to select the cursor type that will be shown when you pass the mouse over the button.

Transition

This option allows you to change the button values with a specific duration.



1. [Home](#)
2. [Layout](#)
3. Button Hover

Button Hover

Font

Family

This option allows you to select the family and the font type that will be used in the button.

Color

This option allows you to select the color that will be shown in the button.

Size

This option allows you to select the button font-size in pixels.

Weight

This option allows you to configure the button **font-weight**.

Shadow

This option allows you to configure the shadow of the button font.

Decoration

This option allows you to select an underline type for the button font.

Style

This option allows you to select the style type that will be applied to the button font.

Border

Width

This option allows you to define the button border width size.

Color

This option allows you to define the button border color.

Style

This option allows you to select the style type that will be applied to the button border.

Radius

This option allows you to configure the border button radius.

Background

Color

This option allows you to define the button background color.

Image

This option allows you to define a image to the button background.

Repeat

This option allows you to configure the repeating for the image selected to the button.

Position

This option allows you to configure the selected image align for the button.

Shadow

This option allows you to configure the shadow of the button background.

Others

Opacity

This option allows you to configure the button opacity.

Height

This option allows you to configure the button height.

Padding

This option allows you to configure the button padding element.

Margin

This option allows you to configure the button margin.

Cursor

This option allows you to select the cursor type that will be shown when you pass the mouse over the button.

Transition

This option allows you to change the button values with a specific duration.

1. [Home](#)
2. [Layout](#)
3. Button Onclick

Button Onclick

Font

Family

This option allows you to select the family and the font type that will be used in the button.

Color

This option allows you to select the color that will be shown in the button.

Size

This option allows you to select the button font-size in pixels.

Weight

This option allows you to configure the button **font-weight**.

Shadow

This option allows you to configure the shadow of the button font.

Decoration

This option allows you to select an underline type for the button font.

Style

This option allows you to select the style type that will be applied to the button font.

Border

Width

This option allows you to define the button border width size.

Color

This option allows you to define the button border color.

Style

This option allows you to select the style type that will be applied to the button border.

Radius

This option allows you to configure the border button radius.

Background

Color

This option allows you to define the button background color.

Image

This option allows you to define a image to the button background.

Repeat

This option allows you to configure the repeating for the image selected to the button.

Position

This option allows you to configure the selected image align for the button.

Shadow

This option allows you to configure the shadow of the button background.

Others

Opacity

This option allows you to configure the button opacity.

Height

This option allows you to configure the button height.

Padding

This option allows you to configure the button padding element.

Margin

This option allows you to configure the button margin.

Cursor

This option allows you to select the cursor type that will be shown when you pass the mouse over the button.

Transition

This option allows you to change the button values with a specific duration.



1. [Home](#)
2. [Layout](#)
3. Button Disabled

Button Disabled

Font

Family

This option allows you to select the family and the font type that will be used in the button.

Color

This option allows you to select the color that will be shown in the button.

Size

This option allows you to select the button font-size in pixels.

Weight

This option allows you to configure the button **font-weight**.

Shadow

This option allows you to configure the shadow of the button font.

Decoration

This option allows you to select an underline type for the button font.

Style

This option allows you to select the style type that will be applied to the button font.

Border

Width

This option allows you to define the button border width size.

Color

This option allows you to define the button border color.

Style

This option allows you to select the style type that will be applied to the button border.

Radius

This option allows you to configure the border button radius.

Background

Color

This option allows you to define the button background color.

Image

This option allows you to define a image to the button background.

Repeat

This option allows you to configure the repeating for the image selected to the button.

Position

This option allows you to configure the selected image align for the button.

Shadow

This option allows you to configure the shadow of the button background.

Others

Opacity

This option allows you to configure the button opacity.

Height

This option allows you to configure the button height.

Padding

This option allows you to configure the button padding element.

Margin

This option allows you to configure the button margin.

Cursor

This option allows you to select the cursor type that will be shown when you pass the mouse over the button.

Transition

This option allows you to change the button values with a specific duration.

1. [Home](#)
2. [Layout](#)
3. Button Selected

Button Selected

Font

Family

This option allows you to select the family and the font type that will be used in the button.

Color

This option allows you to select the color that will be shown in the button.

Size

This option allows you to select the button font-size in pixels.

Weight

This option allows you to configure the button **font-weight**.

Shadow

This option allows you to configure the shadow of the button font.

Decoration

This option allows you to select an underline type for the button font.

Style

This option allows you to select the style type that will be applied to the button font.

Border

Width

This option allows you to define the button border width size.

Color

This option allows you to define the button border color.

Style

This option allows you to select the style type that will be applied to the button border.

Radius

This option allows you to configure the border button radius.

Background

Color

This option allows you to define the button background color.

Image

This option allows you to define a image to the button background.

Repeat

This option allows you to configure the repeating for the image selected to the button.

Position

This option allows you to configure the selected image align for the button.

Shadow

This option allows you to configure the shadow of the button background.

Others

Opacity

This option allows you to configure the button opacity.

Height

This option allows you to configure the button height.

Padding

This option allows you to configure the button padding element.

Margin

This option allows you to configure the button margin.

Cursor

This option allows you to select the cursor type that will be shown when you pass the mouse over the button.

Transition

This option allows you to change the button values with a specific duration.



1. [Home](#)
2. [Layout](#)
3. Container

Container

Into the Scriptcase buttons settings we can find the main layout settings among them we can find the 'Dropdown' and 'Button' options.

If what we need is to customize the dropdown style menu then we can select the option 'Dropdown' then some interesting options are available:

- Container
- Line
- Button Hover
- Button Disabled
- Button Selected
- Button OnClick



The container is the border of the menu, we can change some settings including:

- Border
- Background
- Others

Border

The border offer us some settings:



Width

This option allows you to define the button border width size.

Color

This option allows you to define the container border color.



Style

This option allows you to select the style type that will be applied to the button border.

Radius

This option allows you to configure the border button radius.

Background



Color

This option allows you to define the button background color.

Image

This option allows you to define a image to the button background.

Repeat

This option allows you to configure the repeating for the image selected to the button.

Position

This option allows you to configure the selected image align for the button.

Shadow

This option allows you to configure the shadow of the button background.

Others



Opacity

This option allows you to configure the button opacity.

Padding

This option allows you to configure the button padding element.

Margin

This option allows you to configure the button margin.

Cursor

This option allows you to select the cursor type that will be shown when you pass the mouse over the button.



1. [Home](#)
2. [Layout](#)
3. Line

Line

Into the Scriptcase buttons settings we can find the main layout settings among them we can find the 'Dropdown' and 'Button' options.

If what we need is to customize the dropdown style menu then we can select the option 'Dropdown' then some interesting options are available:

- Container
- Line
- Button Hover
- Button Disabled
- Button Selected
- Button OnClick



The container is the border of the menu, we can change some settings including:

- Border
- Background
- Others

Border

The border offer us some settings:



Width

This option allows you to define the button border width size.

Color

This option allows you to define the container border color.



Style

This option allows you to select the style type that will be applied to the button border.

Radius

This option allows you to configure the border button radius.

Background



Color

This option allows you to define the button background color.

Image

This option allows you to define a image to the button background.

Repeat

This option allows you to configure the repeating for the image selected to the button.

Position

This option allows you to configure the selected image align for the button.

Shadow

This option allows you to configure the shadow of the button background.

Others



Opacity

This option allows you to configure the button opacity.

Padding

This option allows you to configure the button padding element.

Margin

This option allows you to configure the button margin.

Cursor

This option allows you to select the cursor type that will be shown when you pass the mouse over the button.



1. [Home](#)
2. [Layout](#)
3. Button Hover

Button Hover

Into the Scriptcase buttons settings we can find the main layout settings among them we can find the 'Dropdown' and 'Button' options.

If what we need is to customize the dropdown style menu then we can select the option 'Dropdown' then some interesting options are available:

- Container
- Line
- Button Hover
- Button Disabled
- Button Selected
- Button OnClick



The container is the border of the menu, we can change some settings including:

- Border
- Background
- Others

Border

The border offer us some settings:



Width

This option allows you to define the button border width size.

Color

This option allows you to define the container border color.



Style

This option allows you to select the style type that will be applied to the button border.

Radius

This option allows you to configure the border button radius.

Background



Color

This option allows you to define the button background color.

Image

This option allows you to define a image to the button background.

Repeat

This option allows you to configure the repeating for the image selected to the button.

Position

This option allows you to configure the selected image align for the button.

Shadow

This option allows you to configure the shadow of the button background.

Others



Opacity

This option allows you to configure the button opacity.

Padding

This option allows you to configure the button padding element.

Margin

This option allows you to configure the button margin.

Cursor

This option allows you to select the cursor type that will be shown when you pass the mouse over the button.

□

1. [Home](#)
2. [Layout](#)
3. Link Styles

Link Styles

CSS Link Settings

Here we can configure the links that will be shown as Buttons in Scriptcase. Handling various visual aspects related to these links.

Also is allowed to create new CSS Link Styles clicking in the button **New**.

Link

Font

This option allows you to configure the font type that will be used in the link.

Font Size

This option allows you to configure the font size that will be used in the link.

Font Color

This option allows you to configure the link color for the exhibition.

Text-decoration

This option allows you to configure the line personalization type that the link will have.

Cursor

This option allows you to configure the cursor type that will be shown when passing the mouse through the link.

Visited

Font

This option allows you to configure the font type that will be used in the link.

Font Size

This option allows you to configure the font size that will be used in the link.

Font Color

This option allows you to configure the link color for the exhibition.

Text-decoration

This option allows you to configure the line personalization type that the link will have.

Cursor

This option allows you to configure the cursor type that will be shown when passing the mouse through the link.

Active

Font

This option allows you to configure the font type that will be used in the link.

Font Size

This option allows you to configure the font size that will be used in the link.

Font Color

This option allows you to configure the link color for the exhibition.

Text-decoration

This option allows you to configure the line personalization type that the link will have.

Cursor

This option allows you to configure the cursor type that will be shown when passing the mouse through the link.

Hover

Font

This option allows you to configure the font type that will be used in the link.

Font Size

This option allows you to configure the font size that will be used in the link.

Font Color

This option allows you to configure the link color for the exhibition.

Text-decoration

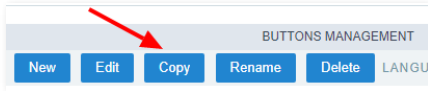
This option allows you to configure the line personalization type that the link will have.

Cursor

This option allows you to configure the cursor type that will be shown when passing the mouse through the link.

Copying a Button Set

To copy a Button set, it is necessary to click on the button **"Copy"** available between the buttons CSS settings toolbar.



After that, will be displayed the copy settings options to the buttons, see below:



New Name

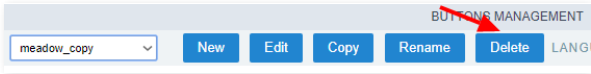
This field allows the user to define the buttons set copy name.

Set Level

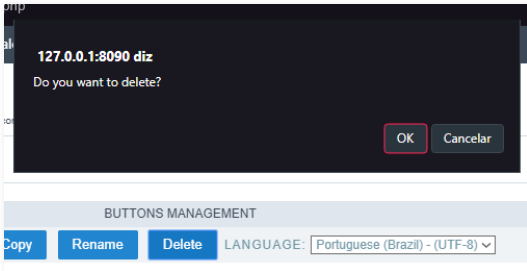
This option allows selecting the level that the buttons set will have. Can choose between: **Scriptcase, Public, Project and User** .

Deleting a Buttons Set

To delete a specific button set, you need just to select the desired set and click on the delete button, as the image:

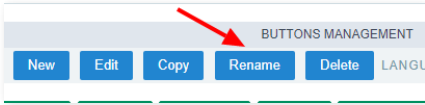


After that, will be displayed a confirmation pop-up for deleting, you will need just click on the button **OK** and the set will be deleted.

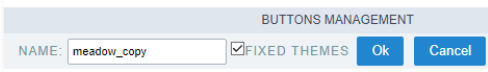


Renaming a Buttons Set

To rename a specific button set, you will need just select the desired set and click on the button **Rename**, as the image:



After that, will be displayed some options, so you can rename the set.



Name

This field allows you to inform the name that will be used in the renamed set.

Fixed Themes

Marked as default, this option allows that when changes the set name, this change will be applied in the projects that use the button set.

In the Templates, it's possible to edit and create models for the Headers, Footers or the User HTML (Grid and Control). These models define the interface structure of the applications. You can create a HTML file and define where the entry data will be, by use curly brackets {} to specify a variable. The User Defined HTML option is used in Grid and Control applications, where it's possible to define all the structure of the report in HTML and associate the fields of the applications with the variables created on the body of the HTML file. With this it's possible to create Control and Grid applications with a different layout, being able to place the fields anywhere on the page.

The screenshot shows the 'HTML Editor' interface. On the left, there is a list of templates under the 'SCRIPTCASE' category, including 'light', 'flat', 'elegant2', 'elegant', 'mainmenu', 'line', 'round', 'thin', 'rounded', 'expanded', and 'default'. A 'Copy' button is next to the 'light' template. On the right, the 'HTML Editor' shows a code editor with the following HTML code:

```

1 <style>
2 #lin1_col1 { padding-left:9px; padding-top:7px; height:27px; overflow:hidden;
3 #lin1_col2 { padding-right:9px; padding-top:7px; height:27px; text-align:right;
4 </style>
5
6 <div style="width: 100%">
7 <div {NM_CSS_FUN_CAB} style="height:11px; display: block; border-width:0px; ">
8 <div style="height:37px; border-width:0px 0px 1px 0px; border-style: dashed;
9 <table style="width:100%; border-collapse:collapse; padding:0;">
10 <tr>
11 <td id="lin1_col1" {NM_CSS_CAB}><span>{LIN1_COL1}</span></td>
12 <td id="lin1_col2" {NM_CSS_CAB}><span>{LIN1_COL2}</span></td>
13 </tr>
14 </table>
15 </div>
16 </div>
17

```

Header and Footer

In the editor for the Header and Footer, you need to create or edit existing models.

To create a new model, there's a standard code to use as a base of all the HTML.

The screenshot shows the 'Template of' dialog box. It has the following fields and buttons:

- Template of: Applications (dropdown)
- Type: Header (dropdown)
- Target: Public (dropdown)
- Template: (empty text field)
- Upload: Browse... (button) No file selected. (text)
- Save (button)
- Cancel (button)

The creation of fields in the HTML needs to be done with curly brackets, for example: {variable}.

Code example:

```

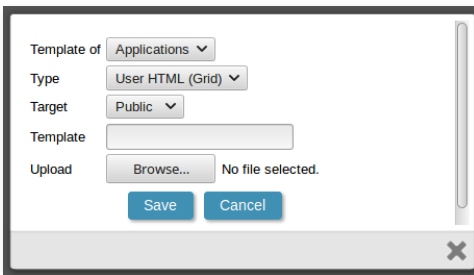
<div style="width: 100%">
<div {NM_CSS_FUN_CAB} style="height:11px; display: block; border-width:0px; "></div>
<div style="height:37px; background-color:#FFFFFF; border-width:0px 0px 1px 0px; border-style: dashed; border-co
<table style="width:100%; border-collapse:collapse; padding:0;">
  <tr>
    <td id="lin1_col1" {NM_CSS_CAB}><span>{LIN1_COL1}</span></td>
    <td id="lin1_col2" {NM_CSS_CAB}><span>{LIN1_COL2}</span></td>
  </tr>
</table>
</div>
</div>

```

These fields are displayed in the application where the template is going to be used.

After saving the template, you need to open the application that you're going to use the template in and go to "Layout > Settings", choose the model for the Header and Footer that you'll use and in "Layout > Header & Footer" you'll associate the created fields in the template with the desired information available by ScriptCase.

User Defined HTML



In the User Defined HTML we can specify where the fields of the applications will be displayed in the HTML file by using curly brackets (variable). Like the example below:

```
<table border="0" cellspacing="0" cellpadding="0">
  <tr>
    <td width="247" height="96" align="left" valign="center">
      <font face="verdana" style="font-size:11px">
        <b>{Name}</b><br>{Address}<br>{City} - {State}<br>
        {ZIP}
      </font>
    </td>
  </tr>
</table>
```

The preview of the code above will be like the following: {Name} {Address} {City}-{State} {ZIP}

Remember that all the content that has curly brackets are considered a field of the application, this is why when you have {Name}, it'll be referencing a field called 'Name' in your application.

In the template we can define a delimiter by using " ":

```
<table border="0" cellspacing="0" cellpadding="0">
  <tr>
    <!-- BEGIN bl2 -->
    <td width="247" height="96" align="left" valign="center">
      <font face="verdana" style="font-size:11px">
        <b>{Name}</b><br>{Address}<br>{City}- {State}<br>
        {ZIP}
      </font>
    </td>
    <!-- END bl2 -->
  </tr>
</table>
```

These parts are defined where the loop is going to start and end. The quantity of delimiters are defined as the "Columns per page" on the Grid application.

e.g.: If within the Grid application the option "Lines Per Page" is set to 3, you'll see the following result.

```
<table border="0" cellspacing="0" cellpadding="0">
  <tr>
    <td width="247" height="96" align="left" valign="center">
      <font face="verdana" style="font-size:11px">
        <b>{Name}</b><br>{Address}<br>{City}- {State}<br>
        {ZIP}
      </font>
    </td>
    <td width="247" height="96" align="left" valign="center">
      <font face="verdana" style="font-size:11px">
        <b>{Name}</b><br>{Address}<br>{City}- {State}<br>
        {ZIP}
      </font>
    </td>
    <td width="247" height="96" align="left" valign="center">
      <font face="verdana" style="font-size:11px">
        <b>{Name}</b><br>{Address}<br>{City}- {State}<br>
        {ZIP}
      </font>
    </td>
  </tr>
</table>
```

The preview of the code above will be the following:

```
{Name}    {Name}    {Name}
{Address} {Address} {Address}
{City}-{State}{City}-{State}{City}-{State}
{ZIP}    {ZIP}    {ZIP}
```

If you have a HTML File created in the standards above, you can include it in ScriptCase by accessing "Layout > HTML Templates > User HTML > New Template". On "New Template" you'll inform the file that you want to import or create a new template by writing the code in the ScriptCase editor.

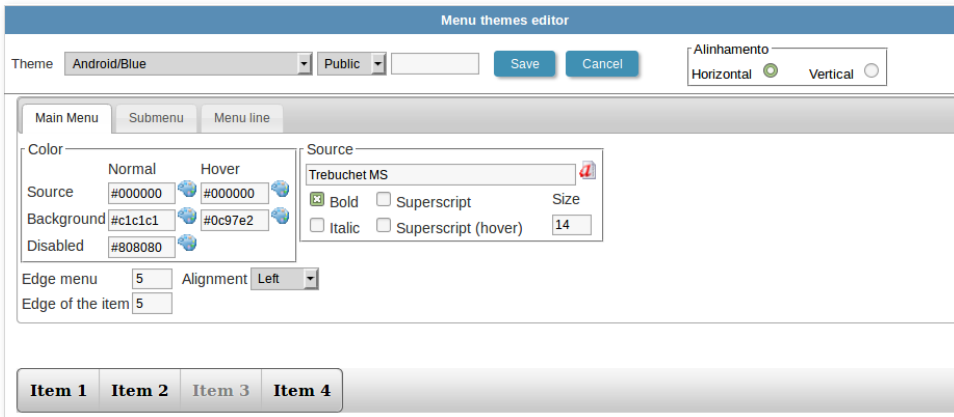
After having the User Defined HTML, the next step is open the application that supports the Template (Control or Grid) and go to "Position of Fields" and associate the fields with the application fields.

The CSS Menu section is used to create and/or edit layout styles to be applied in Menu Applications.



To create a new theme for the menu it's necessary to first select a theme to be used as base of the new theme.

After clicking on "New", to create a new theme for the Menu, you'll have many options to change the colors that are part of the Menu you chosen. Also, you can change the font size of the items.



Before saving the theme that you created, you need to select the access level as Public, Project or User levels.

- **ublic** the theme is available for everyone that uses your Scriptcase installation.
- **roject** the theme is available for the project that it was created in.
- **ser** the theme is available only for the user that created the theme.

After selecting the access level, inform the name of the theme and click on "Save".

While editing the menu icons, you can change the icons of the active and inactive tabs that will be opened in the menu. These icons are linked to the opened applications.

On the page below, we can view the icons divided in blocks for each type of application, so each icon corresponds to an application from ScriptCase.

Select the icons that you desire for each state for the menu tabs, being active or inactive.

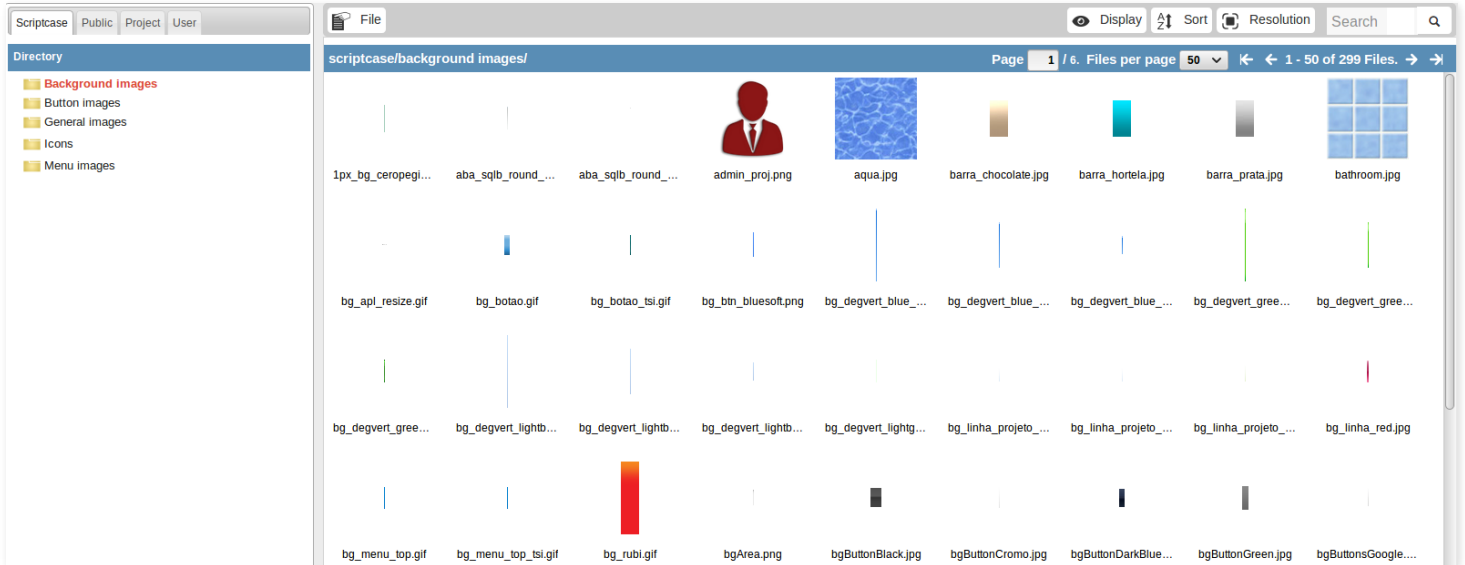
Theme default

Menu tabs icons

Grid	Active <input type="text" value="scriptcase__NM__ico__NM__sc_menu_grid_e.png"/>		Preview
	Inactive <input type="text" value="scriptcase__NM__ico__NM__sc_menu_grid_d.png"/>		
Form	Active <input type="text" value="scriptcase__NM__ico__NM__sc_menu_form_e.png"/>		Preview
	Inactive <input type="text" value="scriptcase__NM__ico__NM__sc_menu_form_d.png"/>		
Control	Active <input type="text" value="scriptcase__NM__ico__NM__sc_menu_control_e.png"/>		Preview
	Inactive <input type="text" value="scriptcase__NM__ico__NM__sc_menu_control_d.png"/>		
Tab	Active <input type="text" value="scriptcase__NM__ico__NM__sc_menu_tabs_e.png"/>		Preview
	Inactive <input type="text" value="scriptcase__NM__ico__NM__sc_menu_tabs_d.png"/>		
Procedures	Active <input type="text" value="scriptcase__NM__ico__NM__sc_menu_grid_e.png"/>		Preview
	Inactive <input type="text" value="scriptcase__NM__ico__NM__sc_menu_grid_d.png"/>		
Menu	Active <input type="text" value="scriptcase__NM__ico__NM__sc_menu_menu_e.png"/>		Preview
	Inactive <input type="text" value="scriptcase__NM__ico__NM__sc_menu_menu_d.png"/>		

The Images Manager is where you'll organize and upload new images that'll be used in your project.

To upload new images, you need to select the scope of which the image will be part, after selecting the folder where the image will be stored at and after that click on File > Upload.



When the image is uploaded, the image's name is modified depending on the scope of which it's part of (public, project or user) and the type of image (background, button, icons, menu or general) following the pattern below:

- ys__NM__ (Public)
- rp__NM__ (Project)
- sr__NM__ (User)

Image Type:

- g__NM__ (background images)
- tn__NM__ (button images)
- mg__NM__ (general images)
- co__NM__ (icons)
- enu_img__NM__ (menu images)

This way, the image called 001.jpg that was added to the public and icon scope will receive the name sys__NM__ico__NM__001.jpg

The image will be in the folder "app/project_name/_lib/img" in the development environment and in the production, will be "/project_name/_lib/img".

After uploading the image to the project, you need to insert the image in the applications that you desire, to do that in the applications desired, go to Applications > Settings > Application images and click on the icon next to the garbage icon to add an image.

The chart themes tool offers the option for you to create different themes for the chart applications of your project. The chart theme allows you to modify some options, such as the background color, font, margin, data, border and shadow.

To create new themes, you need to select an existing theme so that you can use it as a base and on top of it you'll customize it to the way you feel satisfied.

Theme editor Save Cancel

sc_Clean

Chart Type: Chart Column 3D

Weekly Sales Summary

Week	Sales
Week 1	\$14.4K
Week 2	\$8.6K
Week 3	\$25K
Week 4	\$16.7K

Margin: -

Top: px
Right: px
Bottom: px
Left: px
Canvas: px
Value: px

Data: +

Font: +

Background: +

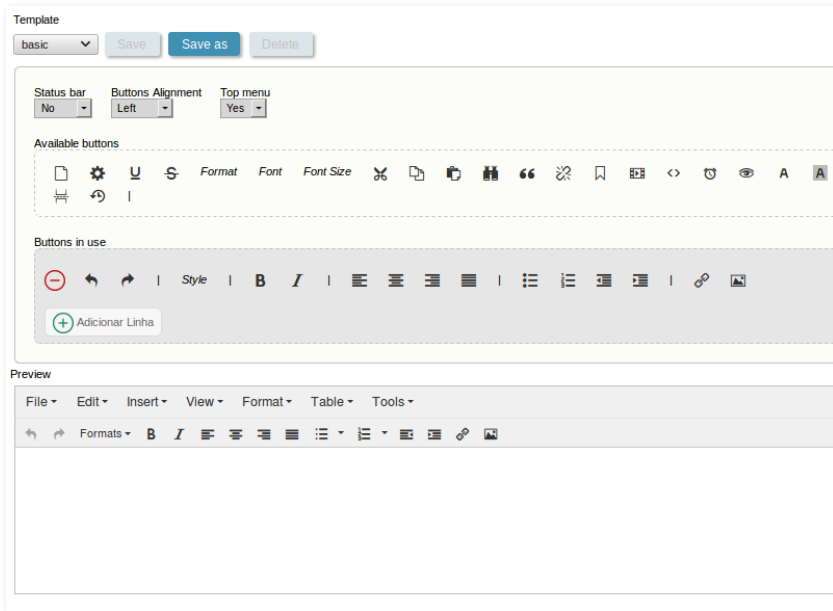
Border: +

Shadow: +

The HTML editor tool offers you the option to modify the toolbar of the html editor field present in the applications.

You can create a new template theme and/or edit an existing one.

To create a new theme, just select an existing one and click on "Save As", to edit an existing theme just select the theme and click on "Save".



ScriptCase offers a complete generator so that you can define the languages and regional settings that your project will support. The locales settings can be accessed via the menu Locales.

Application Language

The language menu for the applications allows you to translate the ScriptCase's default messages, and also create custom messages to be used in different parts of the applications.

To see how to set the Application Language, [click here](#).

Regional Settings

The regional settings allows you to define some parameters of the monetary unit, date and numbers, according to the country or region where your application is going be used. Clicking on Customizing, you can modify the values that'll be used in your applications.

To see how to set the Regional Settings, [click here](#).

The language menu for the applications allows you to translate the ScriptCases default messages, and also create custom messages to be used in different parts of the applications.

Messages

Using the messages menu, it's possible to select and edit messages. The messages are divided in the categories: "ScriptCase Messages" and "Project Messages". New categories can be created using the item "New Folder".

Selecting a lang to be modified, a new page will appear with the columns of each language of your project. You'll inform the value of the message in each language.

▼ MESSAGES

- └ Languages
 - └ Home
 - └ Scriptcase Messages
 - └ JavaScript
 - └ Buttons
 - └ Calendar
 - └ Charts
 - └ Flash Charts
 - └ Log
 - └ Others
 - └ PDF
 - └ Search
 - └ Security
 - └ Validation
 - └ Project messages
 - └ General
 - └ tbl_account
 - └ + New folder

► SEARCH

Index	Default value in English	English	Portuguese/Português (Brasil)
lang_days_frid	Friday	<input type="text" value="Friday"/>	<input type="text" value="Sexta-feira"/>
lang_days_mond	Monday	<input type="text" value="Monday"/>	<input type="text" value="Segunda-feira"/>
lang_days_satd	Saturday	<input type="text" value="Saturday"/>	<input type="text" value="sábado"/>
lang_days_sund	Sunday	<input type="text" value="Sunday"/>	<input type="text" value="domingo"/>
lang_days_thud	Thursday	<input type="text" value="Thursday"/>	<input type="text" value="Quinta-feira"/>
lang_days_tued	Tuesday	<input type="text" value="Tuesday"/>	<input type="text" value="terça"/>
lang_days_wend	Wednesday	<input type="text" value="Wednesday"/>	<input type="text" value="Quarta-feira"/>
lang_mnth_apri	April	<input type="text" value="April"/>	<input type="text" value="Abril"/>
lang_mnth_augu	August	<input type="text" value="August"/>	<input type="text" value="Agosto"/>
lang_mnth_dece	December	<input type="text" value="December"/>	<input type="text" value="Dezembro"/>
lang_mnth_febr	February	<input type="text" value="February"/>	<input type="text" value="Fevereiro"/>
lang_mnth_janu	January	<input type="text" value="January"/>	<input type="text" value="Janeiro"/>
lang_mnth_july	July	<input type="text" value="July"/>	<input type="text" value="Julho"/>
lang_mnth_june	June	<input type="text" value="June"/>	<input type="text" value="Junho"/>
lang_mnth_marc	March	<input type="text" value="March"/>	<input type="text" value="Março"/>
lang_mnth_mayy	May	<input type="text" value="May"/>	<input type="text" value="Maio"/>
lang_mnth_nove	November	<input type="text" value="November"/>	<input type="text" value="Novembro"/>

The regional settings allows you to define some parameters of the monetary unit, date and number according to the country or region where your application is going be used. Clicking on Customize you can modify the values that'll be used in your applications.

Regional Settings ?		
English (United States) Customize...		
EXAMPLES:		
Number	123,456,789.00	-123,456,789.00
Currency unit	\$123,456,789.00	(\$123,456,789.00)
Time	11:13:40 AM	
Date	03/27/2017	
First day	Sunday	

Customizing Regional Settings

In the General group, you can set the writing direction. This setting configures if the characters will be inserted from left-to-right or right-to-left.

General ?	
ATTRIBUTE VALUE	
Writing direction	Left to Right

In the Number group, we have some parameters like the Decimal Symbol, digit grouping and negative format. These parameters are used in the applications that uses the decimal type field.

Number ?	
ATTRIBUTE VALUE	
Decimal symbol	.
Character negative sign	-
Negative number format	-1.1
Character digits grouping	,
Digit grouping	123,456,789

In the Currency Unit group, we have some parameters like the Monetary Symbol, Currency positive format, etc. These parameters are used in the applications that uses the Currency type field.

Currency unit ?	
ATTRIBUTE VALUE	
Symbol	\$
Currency positive format	\$1.1
Currency negative format	(\$1.1)
Decimal symbol	.
Character digits grouping	,
Digit grouping	123,456,789

Within the Time and Date groups are available the parameters of Date and time, that can be used in the fields of the Type Time, Date and Datetime.

Time ?	
ATTRIBUTE VALUE	
Separator	:
Positioning AM/PM	Right with space
Symbol A.M.	AM
Symbol P.M.	PM
Date ?	
ATTRIBUTE VALUE	
Format	mmddyyyy
First day	Sunday
Separator	/

Scriptcase offers a complete generator for your system security through the Security and Log modules. All the Security related to the project users can be found in the Modules menu. Check below how to create these two modules within Scriptcase.

Security Module

It creates an access control with restrictions according to the system users/groups. There are four types of security with different control levels. Scriptcase will create all the necessary database tables and applications to manage the system security according to the type applied.

Scriptcase has five types of security, they are similar, diverging only in how the permissions are applied: **User security, Application security, Group security** and **LDAP security**.

Configure Security Modules is simple and quick. Check the following link to see how to set the module. [click here](#).

Log Module

Using the Log Module, you will be able to track and monitor all actions that the system users will make within the systems developed with the Scriptcase. Together with the Security Module, you will have a complete and effective security system rapidly developed for your applications.

Check the following link to see how to set the Log Module with very simple steps, [click here](#).

Making use of the security modules, you can implement a complete access rules for systems developed by ScriptCase. The process for the security module creation is quick and simple.

Scriptcase has five types of security, they are similar, diverging only in how the permissions are applied.

User Security Type

Under this security type all users have access to all applications. It works only for user authentication. Scriptcase will create one table to store the user and password information and validate the access using a login screen.

In this security type Scriptcase creates these tables:

- **ser** - Login, Password, Name, E-mail, Active, Activation Code and admin privileges
- **ogged users** - Login, login date, Session, IP address

*The table **Logged users** will be created only if the option **Protect Logged users** is checked during the Security Module creation.*

Application Security Type

Under this type Scriptcase will also control login and password as the User Security and also create a restricted access control to the applications or system options according to the logged user. It is possible to define which applications is accessible to each user.

In this security type Scriptcase creates these tables:

- **ser** - Login, Password, Name, E-mail, Active, Activation Code and admin privileges
- **pplication** - Code, Description, Application type
- **ser / Application** - Login, Application Name, Access privileges, Insert privileges, Delete privileges, Update privileges, Export privileges, Print privileges
- **ogged users** - Login, login date, Session, IP address

*The table **Logged users** will be created only if the option **Protect Logged users** is checked during the Security Module creation.*

Group Security Type

This type of security includes the options of User and Application types, however with the Group Security type you will group the users and define the permissions to access the applications or system options according to the groups. One user can belong to one or more groups.

In this security type Scriptcase creates these tables:

- **ser** - Login, Password, Name, E-mail, Active, Activation Code and admin privileges
- **roup** - Description and ID
- **pplication** - Code, Description, Application type
- **ser / Group** - Login, Group ID
- **roup / Application** - Group ID, Application name, Access privileges, Insert privileges, Delete privileges, Update privileges, Export privileges, Print privileges
- **ogged users** - Login, login date, Session, IP address

*The table **Logged users** will be created only if the option **Protect Logged users** is checked during the Security Module creation.*

LDAP Security Type

The Lightweight Directory Access Protocol (LDAP) is an open, vendor-neutral, industry standard application protocol for accessing and maintaining distributed directory information services over an Internet Protocol (IP) network. You can use this protocol within Scriptcase Security module to authentication users in two ways: (**Authentication only**, performs only a simple user authentication, similar to User Security Type and the **Total Control**, where we can define the access by groups, similar to the Group Security Type.

In this security type Scriptcase creates these tables:

- **ser** - Login, Password, Name, E-mail, Active, Activation Code and admin privileges
- **pplication** - Code, Description, Application type
- **ser / Application** - Login, Application Name, Access privileges, Insert privileges, Delete privileges, Update privileges, Export privileges, Print privileges
- **ogged users** - Login, login date, Session, IP address

*The table **Logged users** will be created only if the option **Protect Logged users** is checked during the Security Module creation.*

See how to create each security type:

User security creation

Security Type

The first step is to select the security type you wish.

- User
- Application
- Group
- LDAP
- Only authenticate
- Total control

Connection

Then, you must select the database connection and some extra options to create the tables that make up the security module.

Connection conn_mysql ▾

Use existing tables

Create Tables

Table prefix

Delete tables if they already exist

Protect logged users

Use social networks

Connection

Sets the database connection to create the security tables.

Use existing tables

Let you use existing tables in your database. These tables must have the same fields used by the Scriptcase security module, at least. We recommend you to use this option if you have already used Scriptcase to create the tables previously, to minimize errors.

Create tables

This option creates all security tables in the selected database.

Tables prefix

You can set a prefix to the table names of the Security Module. By default, Scriptcase uses "sec_".

Delete if tables already exist

This option is available only when you select **Create Tables**. By choosing this option, The Security Module drops the tables with the same name and replace them with new tables.

Protect Logged Users

Prevent users to perform simultaneous logins in different sessions.

Tables Link

This step is essential if the option "Use existing tables" is selected. In this case, you must associate the fields from the existing tables to fields of security applications (applications generated by the Security Module).

In these cases, the existing database tables must contain a minimum number of fields so that they can be associated with the fields of security applications.

The required tables for each type of security are described at the beginning of this article.

If you have selected the option "Create tables", in the previous step, it associates the fields automatically.

Settings

In this step, we can define the security module settings, such as User password encryption, the folder to create the applications, the log module, and other settings.

Prefix applications ?

When session expires No action ▾ ?

Encryption ▾ ?

Enable Security Yes No ?

Use Captcha No ▾ ?

Folder ?

Theme Sc9_Rhino ▾ ?

Log ▾ ?

Menu -- Create Menu application -- ▾ ?

Type Menu Tree Menu ▾ ?

Applications Prefix

You can set a prefix to the application names of the Security Module.

When session expires

Define the behaviour when the session expires.

- **o action** - The user continues using the application, but no saves after the session expires.
- **edirect to login after the session expiration** - The application returns to the login after expiring the session.
- **isplay a message that the session has expired** - Shows the message "session has expired" to the user.

Encryption

Use encryption to store the password in the table of users.

Enable Security

Activate the flag **Application Security** for all project applications.

Enable Captcha

It activates the captcha for the login application.

- **o** - Does not display the captcha in the login application.
- **aptcha** - Uses the built-in scriptcase library for captcha display.
- **eCAPTCHA** - Uses Google's reCAPTCHA V2. To configure, [click here](#).

Folder

The project folder name to store the applications generated by Security Module.

Theme

The theme to create the Security Module applications.

Log

This option is available if the project already has a Log Module. [Click here](#) and check out how to create a Log Module

Menu

This option is only available if your project already has a Menu Application. You can associate the existing Menu and include all applications generated by the Security Module to it. If you do not select an existing Menu here, it creates a new Menu Application.

Menu Type

Security Module menu type (option only available if you do not select an existing menu in the previous item)

Login

Here you can set the type and amount of characters allowed for the username and password fields.

User		Password	
Minimum size	<input type="text" value="0"/>	Minimum size	<input type="text" value="0"/>
Maximum size	<input type="text" value="255"/>	Maximum size	<input type="text" value="32"/>
Allowed characters	<input type="text" value="All"/>	Allowed characters	<input type="text" value="All"/>

Minimum size

Minimum amount of characters used by the user.

Maximum size

Maximum size of the characters used by the user.

Characters allowed

Define which characters are allowed when creating a password.

Password Recovery Settings

It defines the password recovery method in the project.

*The password recovery options work only when the SMTP server is configured, please check the **Email settings**.*

- Send email with password
- Reset password and send new one by email
- Send e-mail with a link to set new password

Send the password by e-mail

The system emails the user password just if the SMTP has been correctly set up, and the password is not using encryption.

Reset password and send new one by email

The system resets the password automatically and sends it to the user by email (just if the SMTP has been correctly set up).

Send a link to e-mail with reset option

The system emails the user a link to access an application and reset the password.

New Users

Settings for creating new users in the security system.

The options **activation by email** and **send an email to admin** work only when the SMTP server is configured properly, access **Email settings** to set it up.

Allows the registration of new users

Requires activation by email

Send email to administrator

Default Group

System allows new users registration

This option sets the availability of users registration directly in the login system. If you do not check this option, only users with administrative access can insert new users.

System requires activation by email

This option sets whether the new user must validate his registration by email to activate the account (Configure the email SMTP to use this option)

System requires the user to send an email to admin

This option sets whether the system administrator receives an email whenever a new user is registered. (Configure the email SMTP to use this option)

Email Settings

This option sets the email provider from the system.

The information from this example may change according to the SMTP. We are using the Gmail SMTP for this example.

SMTP Server

SMTP Port

Secure Connection

SMTP User

SMTP Password

SMTP E-mail

SMTP Server

Enter the SMTP server address.

SMTP Port

Enter the SMTP server port. This information must comply with the secure connection option. Use 465 for SSL, 587 for TLS, or 25 for unsafe connection. If you do not inform the port, Scriptcase applies the default one: 25.

Secure Connection

Use SSL or TSL, or leave it blank for insecure connection.

SMTP User

Enter the SMTP User information.

SMTP Password

Enter the SMTP password information.

SMTP E-mail

Enter the SMTP outgoing email.

Logged Users

Defines the system behavior for login protection.

This option is available if you have checked **Protect logged users** during the connection step.

Show logged users Yes No

Brute Force attack protection Yes No

Brute force protection period (in minutes)

Retries number before lock

Display logged users

If you check this option Scriptcase will also create with the Security Module a Grid Application to display a report with all users current logged on the system

Brute Force Attack Protection

Enables/disables the blocking of users after some unsuccessful access attempts.

Brute Force lockout time (in Minutes)

Time, in minutes, that the user will remain inaccessible after several unsuccessful access attempts. (Available only when Brute Force Attack Protection is enabled)

Numbers of attempts before lock

Number of failed access attempts, until the protection is enabled. (Available only when enable protection for brute force attacks)

Enter The First Record

This option inserts the first user into the security tables. This user has administrator privileges to manage the security system and add new users.

Login	<input type="text" value="admin"/>
Password	<input type="password" value="****"/>
Name	<input type="text" value="Admin"/>
E-mail	<input type="text" value="admin@admin.com"/>
Group	<input type="text" value="Administrator"/>

Login

This option sets the system administrator user.

Password

This option sets the system administrator password.

Name

This option sets the system administrator name.

E-mail

This option sets the system administrator user E-mail.

Group

This option sets the group name that receives administrator privileges.

*The group option is available only for **Group Security**, and it is not available for other security module types.*

Add Applications

This option adds the applications already created in the project to the Security Module applications' table.

Checking this option, all applications of this project will be included in your Table of applications.

Application Includes

Save Profile

Using this option, you save all settings during the Security Module creation. It can be used later for other projects.

Save Profile

Profile Name

Target

- Public
- Project
- User

Save Profile

Allows you to save a profile with all the current security module settings.

Name

Profile name. It identifies the profile.

Target

This option sets what developers can use the saved profile afterward.

- **ublic** - Set the security profile available in any project of your Scriptcase.
- **roject** - Set the security profile available only in the current project.
- **ser** - Set the security profile available only to the current Scriptcase user.

Using A Saved Security Profile

You can select a saved profile at the beginning of a new Security Module creation.

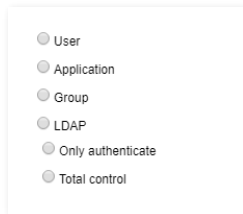
Use Profile

Create New

Application security creation

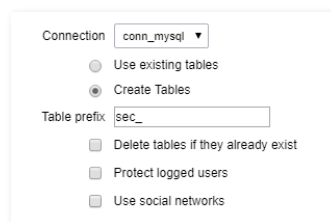
Security Type

The first step is to select the security type you wish.



Connection

Then, you must select the database connection and some extra options to create the tables that make up the security module.



Connection

Sets the database connection to create the security tables.

Use existing tables

Let you use existing tables in your database. These tables must have the same fields used by the Scriptcase security module, at least. We recommend you to use this option if you have already used Scriptcase to create the tables previously, to minimize errors.

Create tables

This option creates all security tables in the selected database.

Tables prefix

You can set a prefix to the table names of the Security Module. By default, Scriptcase uses "sec_".

Delete if tables already exist

This option is available only when you select **Create Tables**. By choosing this option, The Security Module drops the tables with the same name and replace them with new tables.

Protect Logged Users

Prevent users to perform simultaneous logins in different sessions.

Tables Link

This step is essential if the option "**Use existing tables**" is selected. In this case, you must associate the fields from the existing tables to fields of security applications (applications generated by the Security Module).

In these cases, the existing database tables must contain a minimum number of fields so that they can be associated with the fields of security applications.

The required tables for each type of security are described at the beginning of this article.

*If you have selected the option "**Create tables**", in the previous step, it associates the fields automatically.*

Settings

In this step, we can define the security module settings, such as User password encryption, the folder to create the applications, the log module, and other settings.

The screenshot shows a configuration panel for the Security Module with the following settings:

- Prefix applications:
- When session expires:
- Encryption:
- Enable Security: Yes No
- Use Captcha:
- Folder:
- Theme:
- Log:
- Menu:
- Type Menu:

Applications Prefix

You can set a prefix to the application names of the Security Module.

When session expires

Define the behaviour when the session expires.

- **o action** - The user continues using the application, but no saves after the session expires.
- **edirect to login after the session expiration** - The application returns to the login after expiring the session.
- **isplay a message that the session has expired** - Shows the message "session has expired" to the user.

Encryption

Use encryption to store the password in the table of users.

Enable Security

Activate the flag **Application Security** for all project applications.

Enable Captcha

It activates the captcha for the login application.

- **o** - Does not display the captcha in the login application.
- **aptcha** - Uses the built-in scriptcase library for captcha display.
- **eCAPTCHA** - Uses Google's reCAPTCHA V2. To configure, [click here](#).

Folder

The project folder name to store the applications generated by Security Module.

Theme

The theme to create the Security Module applications.

Log

This option is available if the project already has a Log Module. [Click here](#) and check out how to create a Log Module

Menu

This option is only available if your project already has a Menu Application. You can associate the existing Menu and include all applications generated by the Security Module to it. If you do not select an existing Menu here, it creates a new Menu Application.

Menu Type

Security Module menu type (option only available if you do not select an existing menu in the previous item)

Login

Here you can set the type and amount of characters allowed for the username and password fields.

User		Password	
Minimum size	<input type="text" value="0"/>	Minimum size	<input type="text" value="0"/>
Maximum size	<input type="text" value="255"/>	Maximum size	<input type="text" value="32"/>
Allowed characters	<input type="text" value="All"/>	Allowed characters	<input type="text" value="All"/>

Minimum size

Minimum amount of characters used by the user.

Maximum size

Maximum size of the characters used by the user.

Characters allowed

Define which characters are allowed when creating a password.

Password Recovery Settings

It defines the password recovery method in the project.

*The password recovery options work only when the SMTP server is configured, please check the **Email settings**.*

- Send email with password
- Reset password and send new one by email
- Send e-mail with a link to set new password

Send the password by e-mail

The system emails the user password just if the SMTP has been correctly set up, and the password is not using encryption.

Reset password and send new one by email

The system resets the password automatically and sends it to the user by email (just if the SMTP has been correctly set up).

Send a link to e-mail with reset option

The system emails the user a link to access an application and reset the password.

New Users

Settings for creating new users in the security system.

*The options **activation by email** and **send an email to admin** work only when the SMTP server is configured properly, access **Email settings** to set it up.*

- Allows the registration of new users
- Requires activation by email
- Send email to administrator

Default Group

System allows new users registration

This option sets the availability of users registration directly in the login system. If you do not check this option, only users with administrative access can insert new users.

System requires activation by email

This option sets whether the new user must validate his registration by email to activate the account (Configure the email SMTP to use this option)

System requires the user to send an email to admin

This option sets whether the system administrator receives an email whenever a new user is registered. (Configure the email SMTP to use this option)

Email Settings

This option sets the email provider from the system.

The information from this example may change according to the SMTP. We are using the Gmail SMTP for this example.

SMTP Server
 SMTP Port
 Secure Connection
 SMTP User
 SMTP Password
 SMTP E-mail

SMTP Server

Enter the SMTP server address.

SMTP Port

Enter the SMTP server port. This information must comply with the secure connection option. Use 465 for SSL, 587 for TLS, or 25 for unsafe connection. If you do not inform the port, Scriptcase applies the default one: 25.

Secure Connection

Use SSL or TSL, or leave it blank for insecure connection.

SMTP User

Enter the SMTP User information.

SMTP Password

Enter the SMTP password information.

SMTP E-mail

Enter the SMTP outgoing email.

Logged Users

Defines the system behavior for login protection.

*This option is available if you have checked **Protect logged users** during the connection step.*

Show logged users	<input checked="" type="radio"/> Yes <input type="radio"/> No
Brute Force attack protection	<input checked="" type="radio"/> Yes <input type="radio"/> No
Brute force protection period (in minutes)	<input type="text" value="10"/>
Retries number before lock	<input type="text" value="10"/>

Display logged users

If you check this option Scriptcase will also create with the Security Module a Grid Application to display a report with all users current logged on the system

Brute Force Attack Protection

Enables/disables the blocking of users after some unsuccessful access attempts.

Brute Force lockout time (in Minutes)

Time, in minutes, that the user will remain inaccessible after several unsuccessful access attempts. (Available only when Brute Force Attack Protection is enabled)

Numbers of attempts before lock

Number of failed access attempts, until the protection is enabled. (Available only when enable protection for brute force attacks)

Enter The First Record

This option inserts the first user into the security tables. This user has administrator privileges to manage the security system and add new users.

Login	<input type="text" value="admin"/>
Password	<input type="password" value="****"/>
Name	<input type="text" value="Admin"/>
E-mail	<input type="text" value="admin@admin.com"/>
Group	<input type="text" value="Administrator"/>

Login

This option sets the system administrator user.

Password

This option sets the system administrator password.

Name

This option sets the system administrator name.

E-mail

This option sets the system administrator user E-mail.

Group

This option sets the group name that receives administrator privileges.

*The group option is available only for **Group Security**, and it is not available for other security module types.*

Add Applications

This option adds the applications already created in the project to the Security Module applications' table.

<p>Checking this option, all applications of this project will be included in your Table of applications.</p> <p><input checked="" type="checkbox"/> Application Includes</p>

Save Profile

Using this option, you save all settings during the Security Module creation. It can be used later for other projects.

Save Profile

Allows you to save a profile with all the current security module settings.

Name

Profile name. It identifies the profile.

Target

This option sets what developers can use the saved profile afterward.

- **ublic** - Set the security profile available in any project of your Scriptcase.
- **roject** - Set the security profile available only in the current project.
- **ser** - Set the security profile available only to the current Scriptcase user.

Using A Saved Security Profile

You can select a saved profile at the beginning of a new Security Module creation.

Group security creation

Security Type

The first step is to select the security type you wish.

Connection

Then, you must select the database connection and some extra options to create the tables that make up the security module.

Connection

Sets the database connection to create the security tables.

Use existing tables

Let you use existing tables in your database. These tables must have the same fields used by the Scriptcase security module, at least. We recommend you to use this option if you have already used Scriptcase to create the tables previously, to minimize errors.

Create tables

This option creates all security tables in the selected database.

Tables prefix

You can set a prefix to the table names of the Security Module. By default, Scriptcase uses "sec_".

Delete if tables already exist

This option is available only when you select **Create Tables**. By choosing this option, The Security Module drops the tables with the same name and replace them with new tables.

Protect Logged Users

Prevent users to perform simultaneous logins in different sessions.

Tables Link

This step is essential if the option “**Use existing tables**” is selected. In this case, you must associate the fields from the existing tables to fields of security applications (applications generated by the Security Module).

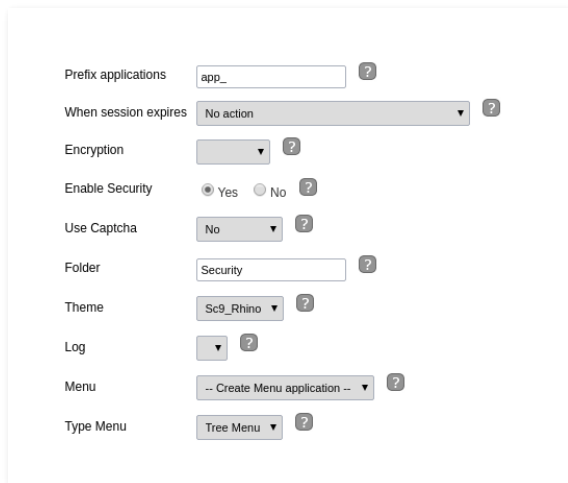
In these cases, the existing database tables must contain a minimum number of fields so that they can be associated with the fields of security applications.

The required tables for each type of security are described at the beginning of this article.

*If you have selected the option “**Create tables**”, in the previous step, it associates the fields automatically.*

Settings

In this step, we can define the security module settings, such as User password encryption, the folder to create the applications, the log module, and other settings.



Applications Prefix

You can set a prefix to the application names of the Security Module.

When session expires

Define the behaviour when the session expires.

- **o action** - The user continues using the application, but no saves after the session expires.
- **edirect to login after the session expiration** - The application returns to the login after expiring the session.
- **isplay a message that the session has expired** - Shows the message “session has expired” to the user.

Encryption

Use encryption to store the password in the table of users.

Enable Security

Activate the flag **Application Security** for all project applications.

Enable Captcha

It activates the captcha for the login application.

- **o** - Does not display the captcha in the login application.
- **aptcha** - Uses the built-in scriptcase library for captcha display.
- **eCAPTCHA** - Uses Google's reCAPTCHA V2. To configure, [click here](#).

Folder

The project folder name to store the applications generated by Security Module.

Theme

The theme to create the Security Module applications.

Log

This option is available if the project already has a Log Module. [Click here](#) and check out how to create a Log Module

Menu

This option is only available if your project already has a Menu Application. You can associate the existing Menu and include all applications generated by the Security Module to it. If you do not select an existing Menu here, it creates a new Menu Application.

Menu Type

Security Module menu type (option only available if you do not select an existing menu in the previous item)

Login

Here you can set the type and amount of characters allowed for the username and password fields.

User	Password
Minimum size <input type="text" value="0"/>	Minimum size <input type="text" value="0"/>
Maximum size <input type="text" value="255"/>	Maximum size <input type="text" value="32"/>
Allowed characters <input type="text" value="All"/>	Allowed characters <input type="text" value="All"/>

Minimum size

Minimum amount of characters used by the user.

Maximum size

Maximum size of the characters used by the user.

Characters allowed

Define which characters are allowed when creating a password.

Password Recovery Settings

It defines the password recovery method in the project.

*The password recovery options work only when the SMTP server is configured, please check the **Email settings**.*

- Send email with password
- Reset password and send new one by email
- Send e-mail with a link to set new password

Send the password by e-mail

The system emails the user password just if the SMTP has been correctly set up, and the password is not using encryption.

Reset password and send new one by email

The system resets the password automatically and sends it to the user by email (just if the SMTP has been correctly set up).

Send a link to e-mail with reset option

The system emails the user a link to access an application and reset the password.

New Users

Settings for creating new users in the security system.

*The options **activation by email** and **send an email to admin** work only when the SMTP server is configured properly, access **Email settings** to set it up.*

- Allows the registration of new users
- Requires activation by email
- Send email to administrator

Default Group

System allows new users registration

This option sets the availability of users registration directly in the login system. If you do not check this option, only users with administrative access can insert new users.

System requires activation by email

This option sets whether the new user must validate his registration by email to activate the account (Configure the email SMTP to use this option)

System requires the user to send an email to admin

This option sets whether the system administrator receives an email whenever a new user is registered. (Configure the email SMTP to use this option)

Email Settings

This option sets the email provider from the system.

The information from this example may change according to the SMTP. We are using the Gmail SMTP for this example.

SMTP Server	<input type="text" value="smtp.gmail.com"/>
SMTP Port	<input type="text" value="587"/>
Secure Connection	<input type="text" value="TLS"/>
SMTP User	<input type="text" value="teste@netmake.com.br"/>
SMTP Password	<input type="password" value="*****"/>
SMTP E-mail	<input type="text" value="teste@netmake.com.br"/>

SMTP Server

Enter the SMTP server address.

SMTP Port

Enter the SMTP server port. This information must comply with the secure connection option. Use 465 for SSL, 587 for TLS, or 25 for unsafe connection. If you do not inform the port, Scriptcase applies the default one: 25.

Secure Connection

Use SSL or TSL, or leave it blank for insecure connection.

SMTP User

Enter the SMTP User information.

SMTP Password

Enter the SMTP password information.

SMTP E-mail

Enter the SMTP outgoing email.

Logged Users

Defines the system behavior for login protection.

*This option is available if you have checked **Protect logged users** during the connection step.*

Show logged users	<input checked="" type="radio"/> Yes <input type="radio"/> No
Brute Force attack protection	<input checked="" type="radio"/> Yes <input type="radio"/> No
Brute force protection period (in minutes)	<input type="text" value="10"/>
Retries number before lock	<input type="text" value="10"/>

Display logged users

If you check this option Scriptcase will also create with the Security Module a Grid Application to display a report with all users current logged on the system

Brute Force Attack Protection

Enables/disables the blocking of users after some unsuccessful access attempts.

Brute Force lockout time (in Minutes)

Time, in minutes, that the user will remain inaccessible after several unsuccessful access attempts. (Available only when Brute Force Attack Protection is enabled)

Numbers of attempts before lock

Number of failed access attempts, until the protection is enabled. (Available only when enable protection for brute force attacks)

Enter The First Record

This option inserts the first user into the security tables. This user has administrator privileges to manage the security system and add new users.

Login	<input type="text" value="admin"/>
Password	<input type="password" value="****"/>
Name	<input type="text" value="Admin"/>
E-mail	<input type="text" value="admin@admin.com"/>
Group	<input type="text" value="Administrator"/>

Login

This option sets the system administrator user.

Password

This option sets the system administrator password.

Name

This option sets the system administrator name.

E-mail

This option sets the system administrator user E-mail.

Group

This option sets the group name that receives administrator privileges.

*The group option is available only for **Group Security**, and it is not available for other security module types.*

Add Applications

This option adds the applications already created in the project to the Security Module applications' table.

Checking this option, all applications of this project will be included in your Table of applications.

Application Includes

Save Profile

Using this option, you save all settings during the Security Module creation. It can be used later for other projects.

Save Profile

Allows you to save a profile with all the current security module settings.

Name

Profile name. It identifies the profile.

Target

This option sets what developers can use the saved profile afterward.

- **ublic** - Set the security profile available in any project of your Scriptcase.
- **roject** - Set the security profile available only in the current project.
- **ser** - Set the security profile available only to the current Scriptcase user.

Using A Saved Security Profile

You can select a saved profile at the beginning of a new Security Module creation.

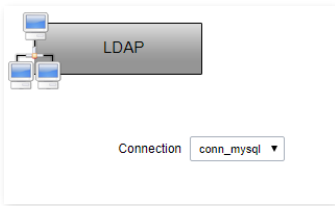
LDAP Creation - Authentication Only

Security Type

The first step is to select the security type you wish.

Connection

Then, you must select the database connection and some extra options to create the tables that make up the security module.



Settings

In this step, we can define the security module settings, such as User password encryption, the folder to create the applications, the log module, and other settings.

Applications Prefix

You can set a prefix to the application names of the Security Module.

When session expires

Define the behaviour when the session expires.

- **o action** - The user continues using the application, but no saves after the session expires.
- **edirect to login after the session expiration** - The application returns to the login after expiring the session.
- **isplay a message that the session has expired** - Shows the message "session has expired" to the user.

Encryption

Use encryption to store the password in the table of users.

Enable Security

Activate the flag **Application Security** for all project applications.

Enable Captcha

It activates the captcha for the login application.

- **o** - Does not display the captcha in the login application.
- **aptcha** - Uses the built-in scriptcase library for captcha display.
- **eCAPTCHA** - Uses Google's reCAPTCHA V2. To configure, [click here](#).

Folder

The project folder name to store the applications generated by Security Module.

Theme

The theme to create the Security Module applications.

Log

This option is available if the project already has a Log Module. [Click here](#) and check out how to create a Log Module

Menu

This option is only available if your project already has a Menu Application. You can associate the existing Menu and include all applications generated by the Security Module to it. If you do not select an existing Menu here, it creates a new Menu Application.

Menu Type

Security Module menu type (option only available if you do not select an existing menu in the previous item)

Login

Here you can set the type and amount of characters allowed for the username and password fields.

Minimum size

Minimum amount of characters used by the user.

Maximum size

Maximum size of the characters used by the user.

Characters allowed

Define which characters are allowed when creating a password.

LDAP

Under this option, you must inform your LDAP server settings.

Server	192.168.254.96
DN	DC=admartini,DC=scriptcase
Port	389
Suffix	@admartini.scriptcase.info

Server

Enter here the LDAP server IP.

DN

Enter the input attributes.

Port

Enter here the port for access to the server. Default port is 389.

Suffix

Enter the user suffix.

Registering The Admin User

This option inserts the first user into the security tables. This user has administrator privileges to manage the security system and add new users.

Login	admin
Password	****
Name	Admin
E-mail	admin@admin.com
<input type="button" value="Test connection"/>	

Login

This option sets the system administrator user.

Password

This option sets the system administrator password.

Name

This option sets the system administrator name.

E-mail

This option sets the system administrator user E-mail.

Save Profile

Using this option, you save all settings during the Security Module creation. It can be used later for other projects.

<input checked="" type="checkbox"/> Save Profile	
Profile Name	security
Target	Public
	Public
	Project
	User

Save Profile

Allows you to save a profile with all the current security module settings.

Name

Profile name. It identifies the profile.

Target

This option sets what developers can use the saved profile afterward.

- **ublic** - Set the security profile available in any project of your Scriptcase.
- **roject** - Set the security profile available only in the current project.
- **ser** - Set the security profile available only to the current Scriptcase user.

Using A Saved Security Profile

You can select a saved profile at the beginning of a new Security Module creation.

The screenshot shows a form with two radio buttons. The first radio button, labeled 'Use Profile', is selected and followed by a dropdown menu containing the text 'security'. The second radio button, labeled 'Create New', is unselected.

LDAP Creation - Total control

Security Type

The first step is to select the security type you wish.

The screenshot shows a list of six radio button options: 'User', 'Application', 'Group', 'LDAP', 'Only authenticate', and 'Total control'. The 'Total control' option is selected.

Connection

Then, you must select the database connection and some extra options to create the tables that make up the security module.

The screenshot shows a form with the following elements: a 'Connection' dropdown menu set to 'conn_mysql'; two radio buttons, 'Use existing tables' (unselected) and 'Create Tables' (selected); a 'Table prefix' text input field containing 'sec_'; and three checkboxes: 'Delete tables if they already exist' (unselected), 'Protect logged users' (unselected), and 'Use social networks' (unselected).

Connection

Sets the database connection to create the security tables.

Use existing tables

Let you use existing tables in your database. These tables must have the same fields used by the Scriptcase security module, at least. We recommend you to use this option if you have already used Scriptcase to create the tables previously, to minimize errors.

Create tables

This option creates all security tables in the selected database.

Tables prefix

You can set a prefix to the table names of the Security Module. By default, Scriptcase uses "sec_".

Delete if tables already exist

This option is available only when you select **Create Tables**. By choosing this option, The Security Module drops the tables with the same name and replace them with new tables.

Protect Logged Users

Prevent users to perform simultaneous logins in different sessions.

Tables Link

This step is essential if the option "Use existing tables" is selected. In this case, you must associate the fields from the existing tables to fields of security applications (applications generated by the Security Module).

In these cases, the existing database tables must contain a minimum number of fields so that they can be associated with the fields of security applications.

The required tables for each type of security are described at the beginning of this article.

If you have selected the option "Create tables", in the previous step, it associates the fields automatically.

Settings

In this step, we can define the security module settings, such as User password encryption, the folder to create the applications, the log module, and other settings.

Prefix applications ?

Enable Security Yes No ?

Use Captcha ?

Folder ?

Theme ?

Log ?

Menu ?

Type Menu ?

Applications Prefix

You can set a prefix to the application names of the Security Module.

When session expires

Define the behaviour when the session expires.

- **o action** - The user continues using the application, but no saves after the session expires.
- **edirect to login after the session expiration** - The application returns to the login after expiring the session.
- **isplay a message that the session has expired** - Shows the message "session has expired" to the user.

Encryption

Use encryption to store the password in the table of users.

Enable Security

Activate the flag **Application Security** for all project applications.

Enable Captcha

It activates the captcha for the login application.

- **o** - Does not display the captcha in the login application.
- **aptcha** - Uses the built-in scriptcase library for captcha display.
- **eCAPTCHA** - Uses Google's reCAPTCHA V2. To configure, [click here](#).

Folder

The project folder name to store the applications generated by Security Module.

Theme

The theme to create the Security Module applications.

Log

This option is available if the project already has a Log Module. [Click here](#) and check out how to create a Log Module

Menu

This option is only available if your project already has a Menu Application. You can associate the existing Menu and include all applications generated by the Security Module to it. If you do not select an existing Menu here, it creates a new Menu Application.

Menu Type

Security Module menu type (option only available if you do not select an existing menu in the previous item)

Login

Here you can set the type and amount of characters allowed for the username and password fields.

User	Password
Minimum size <input type="text" value="0"/>	Minimum size <input type="text" value="0"/>
Maximum size <input type="text" value="255"/>	Maximum size <input type="text" value="32"/>
Allowed characters <input type="text" value="All"/>	Allowed characters <input type="text" value="All"/>

Minimum size

Minimum amount of characters used by the user.

Maximum size

Maximum size of the characters used by the user.

Characters allowed

Define which characters are allowed when creating a password.

Email Settings

This option sets the email provider from the system.

The information from this example may change according to the SMTP. We are using the Gmail SMTP for this example.

SMTP Server	<input type="text" value="smtp.gmail.com"/>
SMTP Port	<input type="text" value="587"/>
Secure Connection	<input type="text" value="TLS"/>
SMTP User	<input type="text" value="teste@netmake.com.br"/>
SMTP Password	<input type="password" value="*****"/>
SMTP E-mail	<input type="text" value="teste@netmake.com.br"/>

SMTP Server

Enter the SMTP server address.

SMTP Port

Enter the SMTP server port. This information must comply with the secure connection option. Use 465 for SSL, 587 for TLS, or 25 for unsafe connection. If you do not inform the port, Scriptcase applies the default one: 25.

Secure Connection

Use SSL or TSL, or leave it blank for insecure connection.

SMTP User

Enter the SMTP User information.

SMTP Password

Enter the SMTP password information.

SMTP E-mail

Enter the SMTP outgoing email.

LDAP

Under this option, you must inform your LDAP server settings.

Server	<input type="text" value="192.168.254.96"/>
DN	<input type="text" value="DC=admartini,DC=scriptcas"/>
Port	<input type="text" value="389"/>
Suffix	<input type="text" value="@admartini.scriptcase.info"/>

Server

Enter here the LDAP server IP.

DN

Enter the input attributes.

Port

Enter here the port for access to the server. Default port is 389.

Sufix

Enter the user suffix.

Logged Users

Defines the system behavior for login protection.

*This option is available if you have checked **Protect logged users** during the connection step.*

Show logged users	<input checked="" type="radio"/> Yes <input type="radio"/> No
Brute Force attack protection	<input checked="" type="radio"/> Yes <input type="radio"/> No
Brute force protection period (in minutes)	<input type="text" value="10"/>
Retries number before lock	<input type="text" value="10"/>

Display logged users

If you check this option Scriptcase will also create with the Security Module a Grid Application to display a report with all users current logged on the system

Brute Force Attack Protection

Enables/disables the blocking of users after some unsuccessful access attempts.

Brute Force lockout time (in Minutes)

Time, in minutes, that the user will remain inaccessible after several unsuccessful access attempts. (Available only when brute Force Attack Protection is enabled)

Numbers of attempts before lock

Number of failed access attempts, until the protection is enabled. (Available only when enable protection for brute force attacks)

Registering The Admin User

This option inserts the first user into the security tables. This user has administrator privileges to manage the security system and add new users.

Form for registering the admin user:

- Login: admin
- Password: ****
- Name: Admin
- E-mail: admin@admin.com
- Test connection button

Login

This option sets the system administrator user.

Password

This option sets the system administrator password.

Name

This option sets the system administrator name.

E-mail

This option sets the system administrator user E-mail.

Add Applications

This option adds the applications already created in the project to the Security Module applications' table.

Form for adding applications:

Checking this option, all applications of this project will be included in your Table of applications.

Application Includes

Save Profile

Using this option, you save all settings during the Security Module creation. It can be used later for other projects.

Form for saving a profile:

- Save Profile
- Profile Name: security
- Target: Public (dropdown menu with options: Public, Project, User)

Save Profile

Allows you to save a profile with all the current security module settings.

Name

Profile name. It identifies the profile.

Target

This option sets what developers can use the saved profile afterward.

- **ublic** - Set the security profile available in any project of your Scriptcase.
- **roject** - Set the security profile available only in the current project.
- **ser** - Set the security profile available only to the current Scriptcase user.

Using A Saved Security Profile

You can select a saved profile at the beginning of a new Security Module creation.

Form for using a saved security profile:

- Use Profile security (dropdown menu)
- Create New

reCAPTCHA

reCAPTCHA is an API provided by Google for forms. It adds security, preventing automatic submission of forms through robots.

reCAPTCHA sample:



1. first, we must request an API Key to activate reCAPTCHA into a Scriptcase application by following the steps below:

To get a **Site key** and **Secret Key** go to the link: <https://www.google.com/recaptcha/admin#list>. See the image:

Register a new site

Label
For example, example.com: Comments page

Choose the type of reCAPTCHA ?

reCAPTCHA V2
Validate users with the "I'm not a robot" checkbox.

Invisible reCAPTCHA
Validate users in the background.

reCAPTCHA Android
Validate users in your android app.

Domains
(one per line)
For example:
example.com
example.net
example.org

Accept the reCAPTCHA Terms of Service.
By accessing or using the reCAPTCHA APIs, you agree to the Google APIs Terms of Use, and to the Additional Terms below. Please read and understand all applicable terms and policies before accessing the APIs.
» reCAPTCHA Terms of Service

Send alerts to owners ?

Register

Label

It is a project identifier to create the reCAPTCHA keys.

Choose the type of reCaptcha

We must choose the option **reCAPTCHA V2**.

Domains

We can insert multiple domains (one per line) to limit the API uses.

1. then, we need to accept the Terms of Service ("Accept the reCAPTCHA Terms of Service").
2. then clicking on **Register**, the page refreshes and shows the integration of reCAPTCHA information. There we can get the **Site Key** and **Secret Key**:

Adding reCAPTCHA to your site

Keys

Site key Use this in the HTML code your site serves to users.	Secret key Use this for communication between your site and Google. Be sure to keep it a secret.
<input type="text"/>	<input type="text"/>

1. now, we can set the Scriptcase application security:

Prefix applications ?

Encryption ?

Enable Security Yes No ?

Use Captcha ?

Site Key ?

Secret Key ?

Folder ?

Theme ?

Log ?

Menu ?

Type Menu ?

Site Key

Key generated by Google after reCAPTCHA project creation.

Secret Key

Key generated by Google after reCAPTCHA project creation.

Watch a video of reCAPTCHA example:

RELATED VIDEOS

 [Control Application](#)

 [Facebook Authentication](#)

 [Twitter Authentication](#)

 [Google Authentication](#)

Using the log module, you can create multiple log schema, each project application can be linked to a different log schema, which will be configured individually. Each project can have one or more log schemes applied to its applications.

Schema list

Access the menu Modules > Log and click **Create / Edit Log Module**.

If you are creating your first log schema project, proceed direct to step three, for the schema Creation/editing.

It lists all log schemes previously created. On this screen, we can manage the schema created.

- **dit Applications** - Edits the relation between the applications and the schemes.
- **ew scheme** - Creates a new Log scheme in the project.
- **reate Application Log** - Creates a report using the log table, that is also used by the project's default scheme.
- **dit Scheme Icon** - Edits the Log scheme.
- **elele Scheme Icon** - Deletes the Log scheme.

SCHEMES	DESCRIPTION	DEAFULT SCHEMA	ACTION
log			

Edit Applications
 New scheme
 Create Application Log

Schema Creation/editing

Access the menu Modules > Log and click **Create / Edit Log Module**.

New Scheme

Name* Name for your schema.

Description Schema description

Connection* Connection to be used to store the log table.

Table* Table to store the log. In case it does not exist, it will be created by Scriptcase automatically.

Delete table if it exists This option will delete (drop) the table if it already exists.

Deafault Schema Set this scheme as default for the project.

Login Variable that stores the login. eg.: [variable]

Default Values

Events Log events that will be saved within tables.

Recording Mode* Only key All Fields Data that will be saved within the Log table.

*required fields

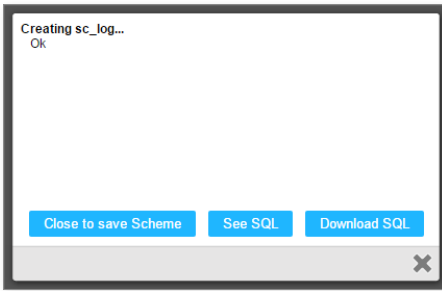
New Schema

- **ame** - Log schema name that will be created. It is a mandatory field.
- **escription** - Short description for the new schema.
- **onnection** - Connection on where the log table will be created. It is a mandatory field.
- **able** - Table name on where the log information will be stored. It is a mandatory field.
- **xisting Table** - This option deletes the table that has the same name in the database and create a new one. If you want to use the existing table to store the log data just leave this option unchecked.
- **efault Schema** - It sets the schema that is being created as the default one for project.
- **ogin** - Global variable used to store the logged user information into the system. This variable must be the same used in the Security Module. By default, Scriptcase sets this variable on both modules as `[usr_login]`.

Default Values

- **vents** - Actions that will be stored in the log table.
- **orage mode** - User data that will be stored in the log table.

A confirmation screen will appear when you save, just click **Close** to leave and save the schema.



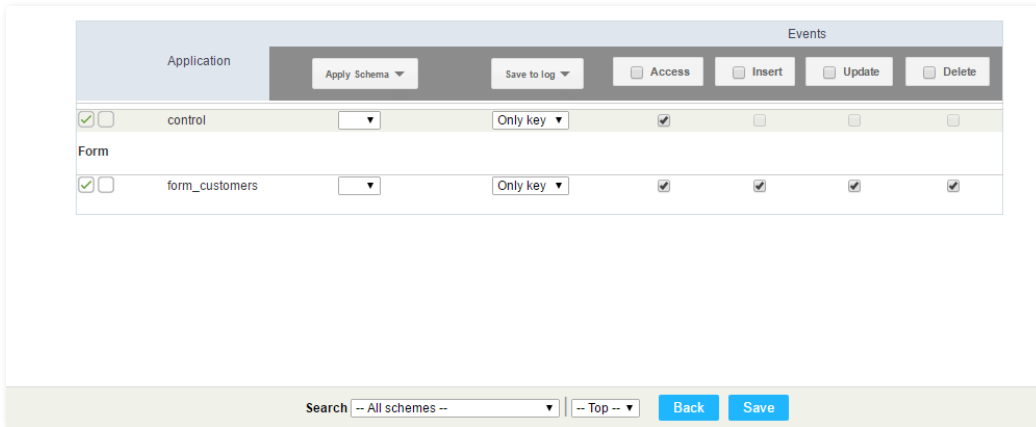
- **lose to save the schema** - Saves the schema you created, and redirects to the applications screen.
- **ee SQL** - Display the SQL used by Scriptcase to create the table 'sc_log' that will be used to store the log events.
- **ownload SQL** - This option will download the SQL used by Scriptcase to create the table 'sc_log'.
- **lose Icon** - This option cancels the schema creation and redirects to a new schema creation.

After the confirmation, you will be redirected to link the log schema to the project applications.

Link Applications

In this step, you must define which log schema will be used in each project application. The list of applications can be filtered by type or by log schema used.

At the top, you can apply actions to all rows or you can select only one action for each row individually.



- **pply schema** - Select the schema that will be used for the applications. (when you select a schema, the other information is filled according to what was previously configured).
- **ore for the Log** - User data that will be stored in the Log table.
- **ccess** - This event stores the information every time the application is accessed.
- **nsert** - This event stores the information when inserting a record.
- **pdate** - This event stores the information when updating a record.
- **delete** - This event stores the information when deleting a record.

Log Report

In this report, you can see all stored information according to the schema selected during the report creation.

ATTRIBUTE	VALUE	DESCRIPTION
Schema	log	Log schema in which the application will be based.
Name	app_grid_log	Name of application that will be generated
Folder	log	Incorrect folder name.
Theme	Sc8_BlueWood	Template application.
Use security	<input checked="" type="radio"/> Yes <input type="radio"/> No	Use safety on application.

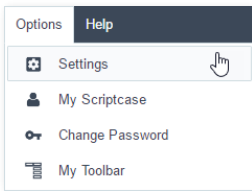
- **chema** - Log schema that the application will be based on
- **ame** - Application name
- **older** - Folder name where the application will be stored
- **heme** - Theme used for the generated application
- **ecurity** - This option attaches a security profile to the log schema.

Gathers the main configurations of ScriptCase. For the users with administrator privileges, it's possible to register the ScriptCase, manage the system users, do backups and restore projects and other configurations.

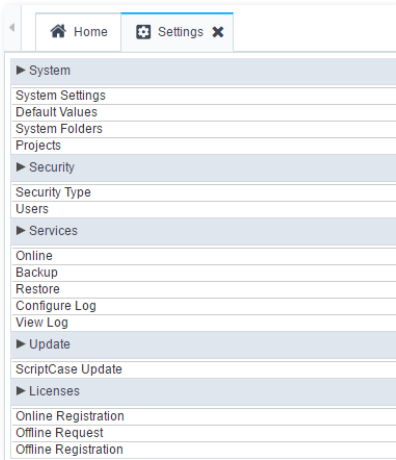
For the other users, without administrator privileges, its possible to define the default language for ScriptCase, customize Scriptcase's toolbar and other options that when done, will be available for the users that did them, for those who won't change these configurations, they'll use the default configurations from ScriptCase.

Access restricted to users with administrative privileges. We have access to the main settings of ScriptCase.

Access Options > Settings.



Left side Menu of the Settings page from ScriptCase.



Below its possible to see some detailed information of each option.

System Settings

Here you can define the settings about the Scriptcase development environment.

System Settings ?	
ATTRIBUTE	VALUE
Language	English ▼
Quantity of compilations in parallel	5
Display the compile-time of the applications individually.	<input type="radio"/> Yes <input checked="" type="radio"/> No
Number of automatic application copies	10
Send an alert if the last backup is older than(days)	10
Show backup alert for	Administrators ▼
Check for ScriptCase update versions after login	Check all days ▼
Performing Scriptcase's database verification.	<input checked="" type="radio"/> Yes <input type="radio"/> No
ScriptCase Session Timeout (min)	30
Time to expire the login cookie (days)	180
Check IP when log on via cookie	<input type="radio"/> Yes <input checked="" type="radio"/> No
Check browser when log on via cookie	<input checked="" type="radio"/> Yes <input type="radio"/> No
Check machine when log on via cookie	<input checked="" type="radio"/> Yes <input type="radio"/> No
Show Generation Status Message	All ▼
Use cache for table fields	<input checked="" type="radio"/> Yes <input type="radio"/> No
SMTP Server	
SMTP Port	25
SSL	<input type="radio"/> Yes <input checked="" type="radio"/> No
SMTP User	
SMTP Password	
SMTP E-Mail	
PROXY Server	
PROXY Gateway	
PROXY User	
PROXY Password	
Use session in database	<input type="radio"/> Yes <input checked="" type="radio"/> No
Show an option on the deployment wizard to store the PHP session variables in a Database	<input type="radio"/> Yes <input checked="" type="radio"/> No
Pass the PHP Session ID in the URL	<input type="radio"/> Yes <input checked="" type="radio"/> No
Timezone	
Update	

Language

Defines the default Language for the projects.

Quantity of compilations in parallel

Defines multiple threads to generate applications. The default value is 5, so it can compile five applications at the same time.

Display the compile-time applications individually

Allows displaying the compilation time for each application individually.

Number of automatic application copies

Scriptcase stores versions of each application automatically. Here, you can set the limit of backup copies it keeps. The older one is overwritten.

Send an alert if the last backup is older than(days)

It allow you to set a number of days to alert when the last Scriptcase backup is obsolete.

Show backup alert for

Defines the user's group to receive the backup alert.

Check for ScriptCase update versions after login

Defines the period to check if there is a new Scriptcase update released.

Performing Scriptcase's database verification

Defines the period to check the backup of ScriptCase's database (Not the projects).

ScriptCase Session Timeout (min)

The time limit for executing PHP scripts during application development. For example, to compile or save applications. (in seconds).

Time to expire the login cookie (days)

The quantity of days Scriptcase keeps the cookies of login.

Check IP when log on via cookie

Scriptcase uses the cookies to validate the IP of the logged user.

Check Browser when log on via cookie

Scriptcase uses the cookies to validate the browser of the logged user.

Check Machine when log on via cookie

Scriptcase uses the cookies to validate the computer configuration of the logged user.

Show Generation Status Message

Allows you to define the action messages of ScriptCase, when it is compiling or running your applications, in the options:

- **ain:** Displays a subgroup of the messages;
- **one:** Does not display the messages;
- **ll:** Enables the display of all the messages.

Use cache for table fields

It makes the applications faster while consulting the data by keeping the fields in the cache.

To recover the ScriptCase password, available at the login, it is necessary to set up the options below correctly.

SMTP Server

You should inform SMTP server address for the provider.

SMTP Port

Port used by the mail server. Use port 465 for security with SSL, port 587 for security with TLS, or port 25 as port without security. By omitting the value, Scriptcase defaults to 25.

SSL

Defines if it uses SSL security or not.

SMTP User

Inform the SMTP server user.

SMTP Password

Inform the SMTP server password.

SMTP E-Mail

Inform the sender e-mail. Who is sending the emails.

PROXY Server

Inform the IP of your Proxy Server.

PROXY Gateway

Inform the Gateway of your Proxy Server.

PROXY User

Inform the proxy user.

PROXY Password

Enter the proxy user password.

Use session in database

It allows us to store the session data into the database.

Show an option on the deployment wizard to store the PHP session variables in a Database

It allows displaying in deployment, an option to store the session into the database.

Pass the PHP Session ID in the URL

It allows passing the PHP Session ID in the URL.

Timezone

It sets the PHP Timezone used in ScriptCase.

Default Values

With this configuration, it is possible to standardize the projects' creation of the applications, increasing the applications' productivity in development. If you wish to change the settings for a specific project, access the menu **Project > Default Values**.

Common Settings

Common Settings	ATTRIBUTE	VALUE
Logo Image		scriptcase__NM_scriptcase.png
Report page rows limit		10
Themes		Sc8_BlueWood
Display Summary		<input type="radio"/> Yes <input checked="" type="radio"/> No
Use template in the HTML Editor		<input type="radio"/> Yes <input checked="" type="radio"/> No
HTML Editor templates		Basic
Initial sort fields		
	text	Ascendant
	date	Descendant
	number	Descendant
	currency	Descendant
Grid		
	Table Width	100
	Table Width Unit	Percent
	Tab a Group By	10px
	Separates the Group By	10px
	Records per page	10,20,50
Form		
	Table Width	100
	Table Width Unit	Percent
Tab		
	Tab Alignment	Left
Menu		
	Horizontal Alignment of the Menu	Left
	Horizontal Alignment of the Items	Left

Logo

The Favicon to represent the project in the browser. It uses the same Favicon for all applications of the project.

Records per page

Amount of records per page.

Theme

The default selected Theme for the projects.

Show Summary

Displays the number of records on the page and the total of the records in the application. This option is able in the application toolbar.

Use template in the HTML Editor

Selecting Yes, it uses the new template of the HTML Editor field. Set the themes through the menu **Layout > HTML Editor Templates**.

HTML Editor template

Defines the template for editing the field HTML Editor.

Initial sort fields

Text

Set the initial sorting for the text fields.

Date

Set the initial sorting for the date fields.

Number

Set the initial sorting for the number fields.

Currency

Set the initial sorting for the currency fields.

Grid

Table Width

It defines the length of the application table. This option is available to select **Pixel** or **Percent** in the table width unit.

Table Width Unit

The unit used to define the application's width. Automatic (Automatic Width according to the fields' size); Pixel (Set the width in pixels. Example: 800 pixels); Percent (Set the width in percentage. Example: 80%)

Tab a Group By

The left margin of the Group By.

Separates the Group

The margin between two Group By.

Lines per page

List of options for the user to select the quantity of the records per page. If you need an option to list all the records, you should add "all" to the options. Ex.: 10,20,30, all

Form

Table Width

It defines the length of the application table. This option is available to select **Pixel** or **Percent** in the table width unit.

Table Width Unit

The unit used to define the application's width. Automatic (Automatic Width according to the fields' size); Pixel (Set the width in pixels. Example: 800 pixels); Percent (Set the width in percentage. Example: 80%)

Tab

Tab Alignment

The alignment of the tabs in the application.

Menu

Horizontal Alignment of the Menu

Menu alignment.

Horizontal Alignment of the Items

Menu item alignment

Template

It defines the default templates of the Header and Footer.

Toolbar Buttons

It defines the default buttons in the toolbar for the new applications.

Directories where information and models will be stored. All the directories are configured automatically in the automatic installation of ScriptCase and in the manual installation when using the **Typical** installation method, making unnecessary any modifications from the user. We recommend not changing any of the fields without contacting the support first.

System Folders ?	
ATTRIBUTE	VALUE
Web Server's Documents Root (absolute path)	C:/Program Files/NetMake/v9/wwwroot/
Application Path	/scriptcase/app
Common Libs Path	/scriptcase/prod
Configuration Folder Path	C:/Program Files/NetMake/v9/wwwroot/scriptcase/conf
Documents Folder	C:/Program Files/NetMake/v9/wwwroot/scriptcase/file/doc
Images Folder	/scriptcase/file/img
Temporary Files Folder	/scriptcase/tmp

- **oot Directory for the Web Server files** - Root of the Web Server, where scriptcase is installed. On IIS servers, the default directory would be "inetpub/wwwroot".
- **pplication Path** - Location where the applications will have stored when created.
- **roduction Path** - Path to where the "prod" directory is located in ScriptCase, in this directory is located all the libraries that the generated applications need to execute.
- **ocuments Directory** - Directory on which the documents used by the applications will be stored.
- **images Directory** - Directory on which the images used in the applications will be stored. The images loaded in the applications, using the upload field (Image File Name) will be stored in this directory.
- **emporary Files Directory** - Directory where the images manipulated by the application is stored temporary. This temporary storage is a condition the the HTML code requires, so that the images can be opened correctly and the specific directory for these images so that they do not get mixed up with the images from the gallery.

Security Type

Select the way how Scriptcase users can access the interface.

Security type

Scriptcase LDAP

Security level

Users

Manage the users and apply the permission individually.

Groups

Manage users through groups by adding users to the groups and setting the permissions to all grupo members.

Group/Users

Manage users and groups, where user permissions can overwrite the permissions of the group.

[Save](#)

Security Type

The method of data storage. We can select the following methods:

Scriptcase

It stores the access data into the ScriptCase's database.

LDAP

It stores the access data into the LDAP Server and Scriptcase goes there to validate the access.

Security Level

Select the type of security. this option is independent of the storage method.

User

It defines the permissions for each user. You must configure them individually.

Group

It defines the permissions for the User Groups. Here, we must associate the users to one or more groups and set the permissions for each group.

User/Groups

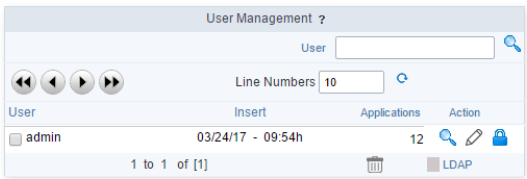
It defines the permissions by the groups and by users. The grant by the user overwrites the group grants. It allows having users existing in the same groups but with different permissions.

The security level option is the same independently of the Security Type. (Scriptcase or LDAP)

Users

User management page of the the system, on it you will view a list with all the users registered on the system. We also have other options, like including new users and editing the existing ones.

*This option will be available on the left side menu when select it in the **Security Type***



- **ser** - Login of the registered Users.
- **nsert** - Date and time of the user's creation.
- **pplications** - Number of applications for each user.

Action

- **etail** - Gives you a detailed view of the user, like it's email, it's privileges, projects and groups of users on which it is part of.
- **dit** - You can change the privileges and the projects that it has access.
- **hange Password** - Change the password for the user.

Below, we will specify some options listed above.

New User

When clicking on the button, **New User** we will be redirected to the page to inform the data for user like login, password, privileges and other things.

- **ser** - Inform a login for the user.
- **-mail** - Defining the user e-mail.
- **ew Password** - Defining the password for the user.
- **onfirm Password** - Confirming the password that is being registered.

Privileges

- **administration** - Permission for the user to access the administrative area of ScriptCase.
- **rojects** - Actions that the user can do when accessing a project.
- **pplication** - Permission to create applications.
- **atabase** - Actions that the user can do with the database configurations in the projects it has access to.
- **roject** - Projects that the user has access.

Next, we will be redirected to a page where you can select which projects will managed by the user. In the example below, I am defining the user **sc** as the administrator of the project Gradebook.

The project administrator, can change various preferences of the project, existing connections.

Editing A User

The editing page is similar to the creation of the users, having the same options of configurations.

Edit User

User:

E-mail:

Administration

Projects

Create Project Data Dictionary
 Run Applications Export
 Edit Libraries Reports
 Locales Deploy

Application

Create Tab Create Blank
 Create Calendar Create Chart application
 Create Grid Application Create Dashboard
 Create Control Create Search
 Create Form Application Create Menu
 Create PDF Report

Database

Database Builder Database Convert
 SQL Builder Connection Management

Project

samples samples1

- **sers** - Users login.
- **-mail** - The e-mail registered for the user.

Privileges

- **administration** - Permission for the user to access the administrative area of ScriptCase.
- **projects** - Actions that user can do when accessing a project.
- **pplication** - Permission to create applications.
- **atabase** - Actions that user can do with the database configurations in the projects it has access to.
- **project** - Projects that the user has access.

Like the procedure for creating a user, we will be redirected to a page where you can select which projects will managed by the user. In the example below, I am defining the user **sc** as the administrator of the project documents_library.

Projects Administrator Link

Select the projects that will be managed by: sc

Projects

samples
samples1

samples1

Changing The Password

On this page, you need to inform the new password for the user.

Change Password

User:

New Password:

Confirm Password:

Delete Users

We can delete more than one user at the same time

When deleting a user that has applications linked to its login, you need to inform if you want to delete the applications or if you want to move them to another user.

User Management ?

User:

Line Numbers:

User	Insert	Applications	Action
<input type="checkbox"/> admin	03/24/17 - 09:54h	12	
<input checked="" type="checkbox"/> sc	03/27/17 - 12:59h	0	

1 to 2 of [2] LDAP

- **delete Application** - All the applications created by the user will be deleted.
- **move Application** - All the applications will be moved to the user informed.

Groups

Managing user groups. You will see a list of the existing groups and some information about them.

This option will only be available on the left side menu when we select the Groups option in the **Security Type**

Name	Description	Users	Projects	#
Group1	Test group		project	Users Edit Delete

Showing 1 to 1 of 1 entries

- **ame** - Name of the groups.
- **escription** - Group Description.
- **sers** - Users that are part of this group.
- **rojects** - Projects that the users of this group have access to.

Actions

- **sers** - Defines the users that will be part of this group.
- **dit** - Edits the group.
- **delete** - Delete the group.

You will see more details below about the existing options.

New Group

Creating the groups, we can define the privileges.

Create Group	
Name	<input type="text"/>
Description	<input type="text"/>
Privileges	<input type="checkbox"/> Admin Privilege
	Project <ul style="list-style-type: none"> <input type="checkbox"/> Create/Edit Projects <input type="checkbox"/> Data dictionary <input type="checkbox"/> Run Applications <input type="checkbox"/> Export <input type="checkbox"/> Libraries <input type="checkbox"/> Reports <input type="checkbox"/> Locales <input type="checkbox"/> Deployment
	Application <ul style="list-style-type: none"> <input type="checkbox"/> Create tab <input type="checkbox"/> Blank <input type="checkbox"/> Create Calendar <input type="checkbox"/> Create Chart <input type="checkbox"/> Create Grid <input type="checkbox"/> Create Dashboard <input type="checkbox"/> Create Control <input type="checkbox"/> Create Search <input type="checkbox"/> Create Form <input type="checkbox"/> Create Menu <input type="checkbox"/> Create ReportPDF
	Connections <ul style="list-style-type: none"> <input type="checkbox"/> Database Builder <input type="checkbox"/> Database Convert <input type="checkbox"/> SQL Builder

- **ame** - Name of the group.
- **escription** - Group Description.
- **dmin Privilege** - Defines if the users that are in the group, will have Administrator Privileges for ScriptCase.

Privileges

- **roject** - Defines if the users from this group will have access to do some actions in the projects.
- **pplication** - Defines the permission to create applications (By type).
- **onnections** - Defines the privileges to the options of the database of the project.
- **rojects** - Defines the projects that will be accessed by the group.
- **dmin projects** - Defines if the group will have Administrator Permission for the project. On this Example, the users from the Group C will be Administrators of the the Gradbook project.

Users

After creating the group, we can define which users will be part of the group.

	<input type="checkbox"/> Create/Edit Connections <input checked="" type="checkbox"/> <input type="checkbox"/>
Projects	<input type="checkbox"/> Agua_e_Gas <input type="checkbox"/> Gas_e_Agua <input type="checkbox"/> Sorteio <input type="checkbox"/> clinica_veterinaria <input type="checkbox"/> onlineShop <input type="checkbox"/> project <input type="checkbox"/> samplesAula <input type="checkbox"/> treinamentoPratico <input checked="" type="checkbox"/> <input type="checkbox"/>
Admin projects	<input type="text"/> Add >> <input type="text"/> Delete <<
<input type="button" value="Cancel"/> <input type="button" value="Save"/>	

Page where you will sync with LDAP. We recommend using the LDAP Administrator user, this way, all the group users of the server will be listed. So, that you can connect with LDAP in ScriptCase and have the LDAP privileges in ScriptCase.

*This option will be available on the left side menu, if you selected the LDAP option in **Security Type***

LDAP Sync

Settings
Groups

LDAP Sync

Server	192.168.254.72
Suffix	@admartini.scriptcase.inf
DN	DC=admartini,DC=scriptc
Username	ralves
Password	*****
Port	389

Test LDAP
Save

- **erver** - IP of the LDAP server (Needs to be setup before proceeding).
- **uffix** - Domain of the server.
- **N** - Domain Component.
- **ser** - Administrator of the Server.
- **assword** - User password informed.
- **ort** - Server Port.

After saving the LDAP server connection, the users and groups listed in the **Users** and **Groups** tabs according to the server. Initially, only the administrator user (The same used for the LDAP connection) will have access to ScriptCase. The other users need to be enabled and their permission needs to be setup also with the groups.

Users

This option is displayed when selecting users in the Security Level, lists all the users from the LDAP server. We can manage the privileges of each user with the projects and options of ScriptCase.

LDAP Sync

Settings
Users

Show 10 entries
Search:

Samaccountname	Name	E-mail	#
Administrator	Administrator	admin@admin.com	Editar Desabilitar
demo	demo		Habilitar
Guest	Guest	guest@guest.com	Habilitar
krbtgt	krbtgt		Habilitar
mack	MAck		Habilitar
ralves	Ronyan		Editar Desabilitar
teste	teste		Habilitar

Showing 1 to 7 of 7 entries
Previous 1 Next

Groups

Option displayed when selecting the group option in the Security Level, lists all the groups from the LDAP server. You need to access this tab to enable the groups and set their privileges.

Sincronizar LDAP			
Configurações		Grupos	
10	resultados por página	Pesquisar	
Nome	Descrição	Usuários	#
Account Operators	Members can administer domain user and group accounts		Habilitar
Administrators	Administrators have complete and unrestricted access to the computer/domain	CN=Domain Admins,CN=Users,DC=admartini,DC=scriptcase,DC=info, CN=Enterprise Admins,CN=Users,DC=admartini,DC=scriptcase,DC=info, CN=Administrator,CN=Users,DC=admartini,DC=scriptcase,DC=info	Habilitar
Allowed RODC Password Replication Group	Members in this group can have their passwords replicated to all read-only domain controllers in the domain		Habilitar
Backup Operators	Backup Operators can override security restrictions for the sole purpose of backing up or restoring files		Habilitar
Cert Publishers	Members of this group are permitted to publish certificates to the directory		Habilitar
Certificate Service DCOM Access	Members of this group are allowed to connect to Certification Authorities in the enterprise		Habilitar
Cryptographic Operators	Members are authorized to perform cryptographic operations.		Habilitar
Denied RODC Password Replication Group	Members in this group cannot have their passwords replicated to any read-only domain controllers in the domain	CN=Read-only Domain Controllers,CN=Users,DC=admartini,DC=scriptcase,DC=info, CN=Group Policy Creator Owners,CN=Users,DC=admartini,DC=scriptcase,DC=info, CN=Domain Admins,CN=Users,DC=admartini,DC=scriptcase,DC=info, CN=Cert Publishers,CN=Users,DC=admartini,DC=scriptcase,DC=info, CN=Enterprise Admins,CN=Users,DC=admartini,DC=scriptcase,DC=info, CN=Schema Admins,CN=Users,DC=admartini,DC=scriptcase,DC=info, CN=Domain Controllers,CN=Users,DC=admartini,DC=scriptcase,DC=info, CN=krbtgt,CN=Users,DC=admartini,DC=scriptcase,DC=info	Habilitar
Distributed COM Users	Members are allowed to launch, activate and use Distributed COM objects on this machine.		Habilitar
DnsAdmins	DNS Administrators Group		Habilitar

Mostrando de 1 até 10 de 37 registros

Anterior 1 2 3 4 Próximo

User / Groups

When selecting a security level of User/Groups, two tabs will be displayed. In this case, the permissions are defined for the group of users and/or individually by user. On this type of security, the permissions of the group and the user are added. So, that can user exist in a group and have different users.

Sincronizar LDAP			
Configurações		usuários	Grupos
10	resultados por página	Pesquisar	
Samaccountname	Nome	E-mail	#
Administrator	Administrator	admin@admin.com	Editar Desabilitar
demo	demo		Habilitar
Guest	Guest	guest@guest.com	Habilitar
krbtgt	krbtgt		Habilitar
mack	MAck		Habilitar
raives	Ronyan		Editar Desabilitar
teste	teste		Habilitar

Mostrando de 1 até 7 de 7 registros

Anterior 1 Próximo

Projects

List of Scriptcase existing projects. You can view all projects details, such as users who have access, the current version of projects, number of applications, among other information.

Project Management ?				
PROJECT	DETAIL	CREATION	APPLICATIONS	ACTION
samples	Samples	09/29/08	309	
samples1	Samples	09/29/08	301	

- **roject** - Project name.
- **escription** - Project description.
- **reation** - Project creation date.
- **pplications** - Number of applications from each project version. 1:33(Versions count: number of applications)
- **ction** - Project details.

By clicking on the magnifying glass next to the number of applications, you can see more details of the project. In addition to some information previously views on the list of projects.

Project Details	
Project	samples
Detail	Samples
Creation	09/29/08 - 09:48h
Connections	conn_example
User	admin
Administrators	admin

- **connections** - Lists of connections of the project.
- **ser** - Users that have access to the project. The users will have the basic permissions, like create applications.
- **administrators** - User/Groups that are administrators of the project. It's possible to create and edit connections in the properties of the project, with this permission.

The permission of each user needs to be setup individually as the group permission also needs to be.

Backup

ScriptCase's backup works similarly as the project export, but the backup does the export of all applications of your ScriptCase. To perform a backup, inform the name to generate the file.

Backup Routine

Backup File Name

tempbkp

Finally, download the generated file and save it in a secure location.

Backup

100%

Download

Restore

This option is similar to Project Import, however using the restore you can import more than one project at a time. When you access the restore you will see a list of backups performed recently, you will be able to restore these backups or to import a backup exported from other Scriptcase installation.

Restore Routine				?
SELECT THE BACKUP FILE.				RESTORE ZIP
BACKUP	CREATED IN	TYPE		
<input type="checkbox"/> bkp_samples	11/17/17 04:54	Global		
<input type="checkbox"/> bkp_OnlineShop	03/13/17 09:00	Global		
<input type="checkbox"/> bkp_OnlineShop	03/13/17 09:00	Global		

- **backup** - Backup generated files.
- **reated in** - Creation date and time of the listed backups.
- **estore** - Backup restore.
- **etail** - Backup details.
- **elete** - Deletes the backup permanently.

To restore another backup, which is not in the list above, select **zip restore** and upload a backup file created by Scriptcase.

After choosing the backup that is going to be restored you will be redirected to a confirmation page. At this moment, you must inform if you want to overwrite duplicate files or keep the existing files in Scriptcase.

Restore Routine			?
BACKUP	CREATED IN	STATUS	
es_20171117-04541420180108011148	p_/bk/sc9_sa:mph	new	
If duplicated :			Do not restore ▼
Process Back			

Log Settings

ScriptCase maintains in its database a log of the main actions done by the users. By default, all the actions are stored in our database. On this option, we can customize the log registry, by marking and unmaking the actions that should be stored.

Log Settings ?

Generate Log

Access	Application	Services	System
<input checked="" type="checkbox"/> Login	<input checked="" type="checkbox"/> Create	<input checked="" type="checkbox"/> Backup	<input checked="" type="checkbox"/> Connections
<input type="checkbox"/> Total	<input checked="" type="checkbox"/> Edit	<input checked="" type="checkbox"/> Restore	<input checked="" type="checkbox"/> Projects
<input checked="" type="checkbox"/> Logout	<input checked="" type="checkbox"/> Delete		<input checked="" type="checkbox"/> Users
	<input checked="" type="checkbox"/> Restore		<input checked="" type="checkbox"/> Data Dictionary
	<input checked="" type="checkbox"/> Export		
	<input checked="" type="checkbox"/> Import		

View Log

This option allows you to view the main actions done by the users, being stored separately by day. On the calendar, we can select the desired date to view the information on the table on the side. The days that have log, have a blueish background color.

? 03/27/17 Log

S	M	T	W	T	F	S	Login	IP	Time	Type	Action	Info	
			1	2	3	4	1	admin	127.0.0.1	09:47:19	Access	Login	en_us
5	6	7	8	9	10	11	2	admin	127.0.0.1	09:47:45	System	Create Connection	Connection: conn_example
12	13	14	15	16	17	18	3	admin	127.0.0.1	09:48:44	Application	Create	Project: samples1 Application: control Type:
19	20	21	22	23	24	25	4	admin	127.0.0.1	10:04:47	Access	Logout	
26	27	28	29	30	31		5	admin	127.0.0.1	10:04:51	Access	Login	en_us
							6	admin	127.0.0.1	10:07:10	Application	Create	Project: samples1 Application: control_1 Type:
							7	admin	127.0.0.1	10:46:16	Access	Login	en_us
							8	admin	127.0.0.1	10:50:21	Application	Edit	Project: samples1 Application: control Type:
							9	admin	127.0.0.1	11:32:07	Access	Login	en_us

March 2017

- **ogin** - Login of the user responsible for the action.
- **P** - IP of the work station that the user did the actions described.
- **ime** - Time when the action was done.
- **ype** - The type of action done.
- **ction** - What action was done.
- **nfo** - Details of the action;

Scriptcase Update

ScriptCase update area. All the update process is done by the INTERNET, using our servers. We recommend using a fast connection.

If the machine where the update will be made doesn't have access to the INTERNET, will be necessary to realize a Scriptcase manual update.

Check our documentation about **Manual Update** according to your operating system:

- [Windows](#)
- [Linux](#)
- [MacOS](#)

Verifying Updates

Firstly, we will verify if there is a new version of ScriptCase available.

ScriptCase Update

Backup (Important)

We strongly recommend that you back up the Scriptcase before upgrading through menu Options > Settings > Services > Backup.

Antivirus

In some cases the antivirus can block the upgrade or leave it a little slower.

Network with restrictions

If your network has a domain constraint scriptcase.net, please contact your network administrator to add an exception.

Scriptcase Update

ScriptCase will check if your ScriptCase version is updated.

[Check ScriptCase Update »](#) ?

Selecting Items For Update

We can later define if the off-line documentation and the prod used by ScriptCase are updated.

ScriptCase Update

Checking ScriptCase Update version...
ScriptCase Update version is up to date...

ScriptCase will analyze your system and search for available updates.

Check ScriptCase Manual Updates
 Check Production Environment Updates

[Search for updates »](#)

Applying Scriptcase Updates

Next, we can visualize the total of updates found.

Search for Available Updates

Analyzing system files...
Done.

Searching for available updates...
Done.

Updates found: 162

ScriptCase will now download the available updates for your system.

[Download and install updates »](#)

After started the update, the process can not be interrupted.

Ending The Update ProcessS

After downloading and updating all the files, you will view a screen with the option to view log of modifications done in the process: [Click here to view ScriptCase update log](#).

ScriptCase was successfully updated.
You need to logout before using ScriptCase.

[Click here to view ScriptCase update log.](#)

[Ok](#)

If you are having problems connecting to our server, check if your firewall is allowing the connections to these links:

- criptcase.com.br
- criptcase.net

On-line registration

For the on-line registration, you must be connected to the INTERNET. There will be an on-line communication between your server and Scriptcase servers. We are going to validate the information that you enter and send you back the confirmation on-line. It's quick and simple.

In order to register your Scriptcase you essentially need to have the account owner information (user, password and serial key). This information is created during the license purchase.

Enter your account details (User login, password and serial key).

To perform the Online Registration, please be sure that the machine where Scriptcase is installed has internet access and with any type of proxy or firewall locks.

Online Registration ?

User:

Password:

Enter your ScriptCase serial number:

User and Password from the license's owner, created during the purchase.
[Forgot your user/password?](#)

[Register](#)

- **ser** - Account user login linked to serial (License owner details, same information used to access **My Scriptcase** options under the website)
- **assword** - Account password
- **erial** - Active and available for registration (open) serial key.

To complete the register, you will need to log out of Scriptcase.

Your copy was successfully registered.
Depending on your Web Server and PHP configuration, you might need to restart it.

Ok

Off-line Registration

Used for cases where the machine where Scriptcase is installed does not have INTERNET access. The off-line registration is done in two steps: Request (request the registration file) and Register (register the file).

In order to register your Scriptcase you essentially need to have the account owner information (user, password and serial key). This information is created during the license purchase.

Request

You can access the Request option under the menu [License > Off-line Request](#)

Inside this option you request an encrypted file containing relevant information to Scriptcase registry (.req file).

The OFFLINE Registration, should be used when the machine with Scriptcase installed, hasn't internet access by proxy lock, firewall or domain restriction.
If you want to try to perform the registry online, make sure your computer is connected to the Internet and that there is nothing that prevents the PHP to perform a remote access. If the ScriptCase can't connect to the remote server, you will be redirected to this page again.

[Try Online Registration](#)

Offline License Request	
User	admin
Password	*****
Enter your ScriptCase serial number.	<input type="text"/>

User and Password from the license's owner, created during the purchase.
[Forgot your user/password?](#)

[RequestIt!](#)

- **ser** - Account user login linked to serial (License owner details, same information used to access **My Scriptcase** options under the website)
- **assword** - Account password
- **erial** - Active and available for registration (open) serial key.

After completing the request, you need to access the website through the link <https://www.scriptcase.net/licenserequest/> to upload the file and get the registry file

File generation form (.lic)

Type the serial: *

Select the license's file (.req) [Upload](#)

[CONTINUE](#)

Terms:
Each license can only work on a single computer, getting under my full responsibility to guarantee compliance with that condition.

Enter the serial you want to use for the registry and upload the file **.rec** delivered by Scriptcase during the file request step. You will receive another file called: **.lic**, download this file and go back to Scriptcase registration step.

Registration

Back to Scriptcase, please access: [Licenses > Off-line Registration](#)

Reenter the account information and the serial key, upload the **.lic** downloaded from the website. S

To register your Scriptcase using the Offline process, you must have the file (.lic) generated by the link: <https://www.scriptcase.net/licenserequest/>

Offline License Registration	
User	admin
Password	*****
Enter your ScriptCase serial number.	<input type="text"/>
<input type="button" value="Escolher arquivo"/> Nenhum arquivo selecionado	

[Registrar](#)

- **ser** - Same login used to generate the file `.rec` (License owner details, same information used to access **My Scriptcase** options under the website)
- **assword** - Account password
- **erial** - Active and available for registration (open) serial key.
- **pload** - Upload the file `.lic` downloaded from the website.

To complete the register you will need to log out of Scriptcase.

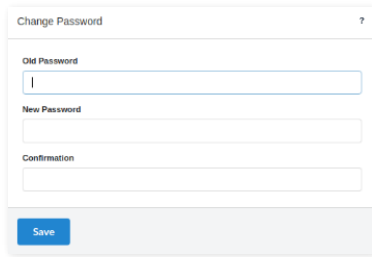
Interface where you can define the ScriptCase settings. All the users have access to this section, and the changes done are tied to the login(user) that you are currently using. This way each user can have their own settings for ScriptCase.

- **language** - Scriptcase interface language. When informing the language in **My ScriptCase**, the selected language at the login won't be considered.
- **Security Resources** - When Uncheck these options, the security settings won't be considered when running the applications in the development environment. When deploying the applications, they will respect the security settings normally.
- **Initial Information** - Defines the information that will be displayed on the application listing.
- **Show Friendly URL** - Defines the display of the field to inform the Friendly URL in the application.
- **Main Menu Interface** - Defines the width of the left side menu of the home page, where the applications are listed and the option of incrementation.
- **Application Settings Interface** - Defines the width of the left side menu of the applications settings and the option of incrementation.
- **Auto Save** - Allows the modifications to be saved automatically.
- **Display INSERT CODE on events** - Defines if the examples will be displaying in the events of the applications.
- **Convert indentation to tab** - Allows to use "tab" hotkey to indent your code or line selected.
- **Show folders at home** - Allows for the sub-folders to be listed with the applications, when selecting a folder on the side menu in the home of the project.
- **Hotkeys** - This option allows to enable/disable Scriptcase hotkeys. [Click Here](#) to view the hotkeys list.

Interface where you can change the password of your user. We always recommend using passwords with a certain level of complexity, to prevent hacks of brute force.

Inform your current password, and the new password twice just to confirm it.

For the option of password recovery to work, the administrator need to setup the SMTP configurations correctly in ScriptCase's settings.



Change Password ?

Old Password

New Password

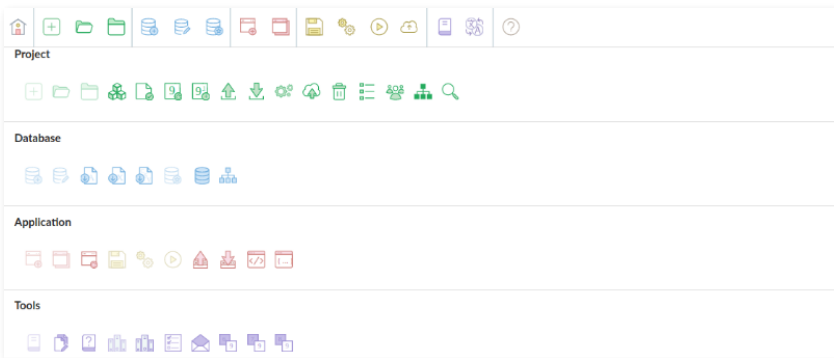
Confirmation

Save

- **Old Password** - Inform your current password.
- **New Password** - Inform your new password.
- **Confirmation** - Confirm the new password.

Allows each user to customize their toolbar, by including or removing unnecessary items for your usage, giving you a better experience while developing the applications. This option is tied to the user that did the modification, this way each user can modify their own toolbar the way they desire.

The icons are divided in categories. To include an item to the toolbar, we just need to drag it to the toolbar.



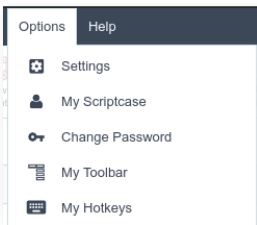
- **oolbar** - Icons that'll be displayed on the toolbar of ScriptCase.

Buttons

- **ave** - Saves the modifications done to the toolbar.
- **dd Separator** - Adds a separator to the toolbar, after the last item of the current setup.
- **reate item** - Creates a new item in the toolbar. Inform the URL, icon and description of the new item.
- **eturn to default** - Restores the toolbar to its default configuration.

Allows you to change the default Hotkeys of your Scriptcase interface, offering to you more practicality and customization to your development.

To change the default hotkeys, you just need access the menu **Options > My Hotkeys**.



Will be open the hotkeys and the setting options, as the image below:

Edit Keyboard Shortcuts

Clear Restore Default +

ACTION:	KEYBINDING:
Save application	CONTROL + S
ACTION:	KEYBINDING:
Save application	⌘ + S
ACTION:	KEYBINDING:
Save application	F7
ACTION:	KEYBINDING:
Generate source	CONTROL + B
ACTION:	KEYBINDING:
Generate source	⌘ + B
ACTION:	KEYBINDING:
Generate source	F8

Upload settings Download settings Save ?

- **lear:** Clears all predefined settings by Scriptcase or by user.
- **estore Default:** Restores all the default shortcuts that was changed or erased.
- **ction:** Selects the action that will be made when press the selected keybind.
- **dd "+":** Adds a new action in the keybind list.
- **eybinding:** Selects the keybinds that will be responsible for execute the chosen action.
- **pload Settings:** Realizes the importation of the keybind settings exported.
- **ownload Settings:** Realizes the download of the keybind settings in use.

A deploy is the final thing you will need to do with your project, so that your applications can be accessed by in the production environment. Basically, it is the development's final phase, where the ScriptCase user will find the help to publish the project on a WEB server, this server being correctly setup to guarantee the perfect functionality of the applications.

Scriptcase offers two ways to deploy your project, both of them are detailed below:

We recommend that the production's PHP version environment, needs to be the same as the Development environment, due to compatibilities of the functions between the versions.

Typical Deploy

It is the type of deploy which the production environment is automatically setup, with all the common libraries used in the applications. In case you need to customize the deploy directories, like the image directory or the temporary directory for example, this type of deployment is not recommended, considering that these directories are setup by ScriptCase default, not allowing customizing their locations, to deploy with the liberty to choose the directories' location you will need to do the "Advance" deployment.

To proceed with the Typical Deploy [Click Here](#).

Advanced Deploy

It is a deploy that gives the ScriptCase User the possibility to choose the default directories for the project, e.g., Image Directory, Document Directory, Prod (Common Libraries) Directory and etc. This type of deployment is recommended if you need to specify the directories for the WEB Server, to be used by your projects.

To proceed with the Advanced Deploy [Click Here](#).

Pre-Requirements

NOTE: Verify [here](#) the functions that need to be enabled in your PHP for the good functioning from Scriptcase generated applications.

Verify below the requirements for the generated applications by Scriptcase. To guarantee that applications can run in Production Environment.

1 - Web Server to deployment (ex: Apache, IIS).

2 - Correct PHP Version for applications developed in Scriptcase.

If you deploy projects developed in Scriptcase with PHP 7.0, the applications you create are homologated to work only with these versions of PHP:

- HP 5.6
- HP 7.0

If you deploy projects developed in Scriptcase with PHP 7.3, the applications you create are homologated to work only with these versions of PHP:

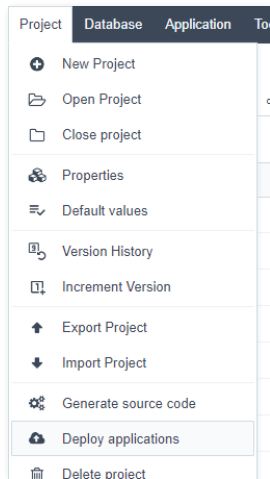
- HP 5.6
- HP 7.0
- HP 7.1
- HP 7.2
- HP 7.3

3 - Access to the project database that is going to be used in the production environment.

This type of deployment is for users that want the Production Environment (Common Libraries) with the default settings from ScriptCase, the basic setup is done already by ScriptCase. We will show the entire process of deployment in this documentation.

Project Deployment

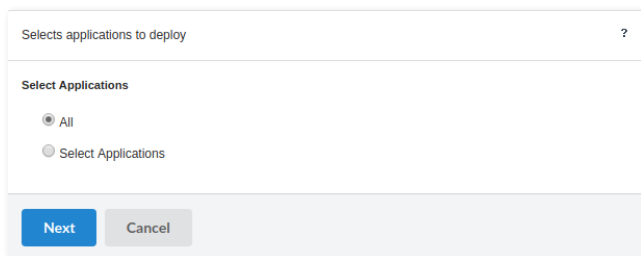
Access your project and choose the option **"Project -> Deploy applications"**, available on the top menu.



After clicking on the Deploy applications button, you will have the option to deploy all the applications from the project or you can select the applications that you want to deploy.

Select All Applications

When select this option, all the applications of the project will be deployed.



Select The Applications Individually

This option allow us to select the applications that will be deployed.

Selects applications to deploy ?

Select Applications

All
 Select Applications

List Of Available Applications

During this process, the Scriptcase list all the Applications in the project so you can select the applications that you wish to deploy.

Applications ?

Visualization Per Type Per Folder

Blank

testes

Calendar

calendar_valores calendar_valores_1

Chart

chart_valores

Grid

ap1p_grid_sec_users app_grid_sec_apps app_grid_sec_groups
 app_grid_sec_users app_grid_sec_users_groups grid_valores

Control

ap1p_Login ap1p_change_pswd ap1p_retrieve_pswd
 app_Login app_change_pswd app_retrieve_pswd
 app_sync_apps

Search

app_search_sec_groups search_valores

Per type	Selecting this option, you can see the list of applications grouped by your respective type.
Per folder	Selecting this option, you can see the list of applications grouped by your respective folder.

Choose The Type Of Deploy That You Will Use (Typical).

Next, we need to choose the type of desired deployment.

Applications Deployment ?

What type of deployment would you like to use?

Typical (recommended)
 Advanced

Setting The Typical Deployment

We must define which application will be the deployment initial application. In this step, we also define if we will send the common libraries with our project.

Applications Deployment ?

Deploy with the common libraries
 Deploy with the common files(CSS, buttons, images, messages)

Publish with FusionChart maps

Publishing server operating system

Windows
 Linux 32 bits
 Linux 64 bits
 OSX / Mac OS

Choose the initial application teste ▼

It is necessary to select the common libraries, which are the files responsible for the Production Environment. It is necessary to select the common files, for the images, css and custom messages to be sent at the moment of the deploy.

Deploy with the Common Libraries(Prod)	Contains the used libraries by our applications and also the informations of connections used by the project.
Deploy with the common files	This option is obligatory for a project deployment, for the images, css and customized messages be send in the moment of deploy generation.
Publish with FusionChart Maps	Checking this option, the referred files to the FusionCharts library will be added in the deployment file.
Publishing server operating system	This option is obligatory for a project deployment, when you select the operating system, the Scriptcase will create the correct structure for your deployment.
Choose the Initial Application	Defines the system initial application.

Generated Files

Generating The Files To Deploy

In this step we must select the way that we wish the applications will be generated.

Generate ZIP or Tar.Gz with applications

The Scriptcase will generate a file **Zip** or **Tar.Gz**. In the end of this process, will be displayed a report of the deployed applications with the download link for download the project.

NOTE: The file **ZIP** or **Tar.Gz** generated, must be decompressed only in the deployment server, to avoid that the files be corrupted.

Deploy on a server directory

Deploys the project in the informed directory. We must inform the absolute path to the deployment directory.

NOTE: This option should be used in the case that the deploy is going to be in the same sever where ScriptCase is installed, otherwise this option is not valid.

Deploy on FTP server

This option generates the files directly in the FTP remote server.

FTP server	Server where you will deploy your project. Inform the domain or the server's IP.
User	User for the FTP server in the production, remembering that the user need to have permission to write in the server folder, otherwise it will not work.
Password	Password for the FTP user.
FTP Folder	Server directory where the project will be deployed in.

NOTE: To use this option is necessary that you have a server that provides the FTP access and an user with write permissions.

Deploy on a SFTP server

Deploys the application automatically on a remote SFTP server.

Applications Deployment ?

Create ZIP with applications
 Generate Tar.Gz with applications
 Deploy on a server directory
 Deploy on FTP Server
 SFTP

- SFTP Server:
- Port:
- SFTP User:
- SFTP Password:
- SFTP Path (Full path):

Back Next Help

SFTP server	Server where you will deploy your project. Inform the domain or the server's IP.
User	User for the SFTP server in the production, remembering that the user need to have permission to write in the server folder, otherwise it will not work.
Password	Password for the SFTP user.
SFTP Folder	Server directory where the project will be deployed in.

NOTE: To use the **SFTP** option is necessary that the user have permission to write on the server folder.

At the end of the process you will receive a report with all the applications deployed and a download link of the project ready to be published.

NOTE: The file **ZIP** or **Tar.Gz** generated, must be decompressed only in the deployment server, to avoid that the files be corrupted.

Applications Deployment

100%

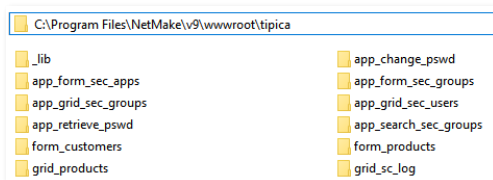
Deployment finished

Click to download your applications in ZIP: [Sorteio_20180112091037.zip](#)

Applications	Result	Processing Time
testes	OK	0.02
calendar_valores	OK	0.06
calendar_valores_1	OK	0.06
chart_valores	OK	0.04
app1p_grid_sec_users	OK	0.05
app_grid_sec_apps	OK	0.04
app_grid_sec_groups	OK	0.04
app_grid_sec_users	OK	0.04
app_grid_sec_users_groups	OK	0.04

Server Directories

In the server root was created a folder(typical) where we will decompress the deployment files.



In the end of the compression process, we must access your project through the browser.

NOTE: In the first access we must configure the connections in [Production Environment](#).

Connections Create

The database connection that your application use was not found. You need to access the production environment and create the connection.

Connections not found: conn_mysql

[Click here to create the connections now.](#)

Pre-Requirements

NOTE: Verify [here](#) the functions that are needed to be enabled in your PHP for the good functioning from Scriptcase generated applications.

Verify below the requirements for the generated applications by Scriptcase. To guarantee that applications can run in Production Environment.

1 - Web Server to deployment (ex: Apache, IIS).

2 - Correct PHP Version for applications developed in Scriptcase.

If you deploy projects developed in Scriptcase with PHP 7.0, the applications you create are homologated to work only with these versions of PHP:

- HP 5.6
- HP 7.0

If you deploy projects developed in Scriptcase with PHP 7.3, the applications you create are homologated to work only with these versions of PHP:

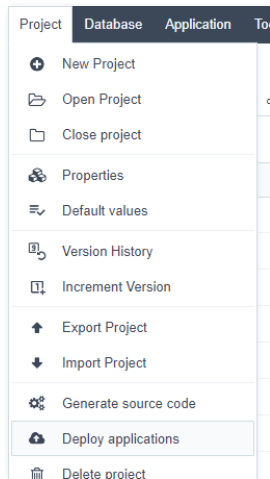
- HP 5.6
- HP 7.0
- HP 7.1
- HP 7.2
- HP 7.3

3 - Access to the project database that is going to be used in the production environment.

Different from the typical deploy, where the directories' setup is done from ScriptCase using the default settings, the advanced deployment allows you to choose the directories according to the server's architecture where it is going to be deployed.

Project Deployment

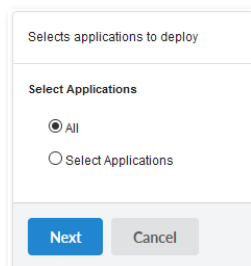
Access your project and choose the option **"Project -> Deploy applications"**, available on the top menu.



After clicking on the Deploy applications button, you will have the option to deploy all the applications from the project or you can select the applications that you want to deploy.

Select All Applications

When you select this option, all the applications of the project will be deployed.



Selecting the Deployment Template

Create a deployment template to save the defined settings. Or, you can use an existing template to reuse the established settings in your project.

Applications Deployment

Use an existent Template
 Create new deployment template

Template Name
news

Back Next

Template Name This option is necessary to inform the deployment template name that will be used in the project. This option can be reused in future deployments.

Setting the Advanced Deploy

We must define which application will be the project initial application. In this step we also define if we will send the FusionCharts Maps in the deployment file.

Applications Deployment

Theme news

Deploy with the common files(CSS, buttons, images, messages)
 Publish with FusionChart maps

Publishing server operating system

Windows
 Linux 32 bits
 Linux 64 bits
 OSX / Mac OS

- Common Libraries Folder
/scriptcase/prod
- Images Folder
/scriptcase/file/img
- Temp Folder
/scriptcase/tmp
- Documents Folder
/home/raptorga/public_html/scriptcase/file/doc

Choose the initial application
app_login

Back Next

NOTE: It is necessary to select correctly the deployment server operating system, allowing the correct generation of the folders structures for deployment.

Deploy with the common files	This option is obligatory for a project deployment, for the images, CSS and customized messages be send in the moment of deploy generation.
Publish with FusionChart Maps	Checking this option, the referred files to the FusionCharts library will be added in the deployment file.
Publishing server operating system	This option is obligatory for a project deployment, when you select the operating system, the Scriptcase will create the correct structure for your deployment.
Common Libraries Folder	It is necessary define the main directory where will be deployed the production environment. By default, the Scriptcase recommends to use: <code>/scriptcase/prod</code> .
Images Folder	It is necessary define the directory that will be stored the images in the production environment. By default, the Scriptcase recommends to use: <code>/scriptcase/file/img</code> .
Temp Folder	It is necessary define the directory that will be stored the temporary files in the production environment. By default, the Scriptcase recommends to use: <code>/scriptcase/tmp</code> .
Documents Folder	It is necessary define the directory that will be stored the documents in the production environment. By default, the Scriptcase recommends to use: <code>/home/\$user/public_html/scriptcase/file/doc</code> . Is this option is necessary to inform the absolute path to the documents folder.
Choose the Initial Application	Defines the system initial application.

Select The Applications Individually

This option allow us to select the applications that will be deployed.

Selects applications to deploy

Select Applications

All

Select Applications

List of available applications

During this process, the Scriptcase list all the Applications in the project so you can select the applications that you wish to deploy.

Select Applications

Applications ?

Visualization Per Type Per Folder

Grid

news_categorias_cad_cons news_detalhes_noticia news_editorial_cons

news_editorial_list_categ_cons news_editorial_list_categ_cons_new news_editorial_por_sessao_cons

news_noticias_caixa_saída_cons news_noticias_editar_publicadas news_noticias_publicar_cons

news_usuarios_cad_cons

Control

api_inicial news_troca_senha_controle

Control

login

Form

news_categorias_cad_frm news_config_sis_frm news_noticias_publicar_frm

news_noticias_redigir_frm news_usuarios_cad_frm

Menu

menu

Per type	Selecting this option, you can see the list of applications grouped by your respective type.
Per folder	Selecting this option, you can see the list of applications grouped by your respective folder.

Choose The Type Of Deploy

Next, we need to choose the type of desired deployment.

Applications Deployment ?

What type of deployment would you like to use?

Typical (recommended)

Advanced

Create a deployment template to save the defined settings. Or, you can use an existing template to reuse the established settings in your project.

Applications Deployment ?

Use an existent Template

Create new deployment template

Template Name

Template Name	This option is necessary to inform the deployment template name that will be used in the project. This option can be reused in future deployments.
----------------------	--

Setting the Advanced Deploy

We must define which application will be the project initial application. In this step we also define if we will send the FusionCharts Maps in the deployment file.

Applications Deployment ?

Theme news

Deploy with the common files(CSS, buttons, images, messages)

Publish with FusionChart maps

Publishing server operating system

Windows

Linux 32 bits

Linux 64 bits

OSX / Mac OS

- Common Libraries Folder
- Images Folder
- Temp Folder
- Documents Folder

Choose the initial application

NOTE: It is necessary to select correctly the deployment server operating system, allowing the correct generation of the folders structures for deployment.

Deploy with the common files	This option is obligatory for a project deployment, for the images, CSS and customized messages be send in the moment of deploy generation.
Publish with FusionChart Maps	Checking this option, the referred files to the FusionCharts library will be added in the deployment file.
Publishing server operating system	This option is obligatory for a project deployment, when you select the operating system, the Scriptcase will create the correct structure for your deployment.
Common Libraries Folder	It is necessary define the main directory where will be deployed the production environment. By default, the Scriptcase recommends to use: <code>/scriptcase/prod</code> .
Images Folder	It is necessary define the directory that will be stored the images in the production environment. By default, the Scriptcase recommends to use: <code>/scriptcase/file/img</code> .
Temp Folder	It is necessary define the directory that will be stored the temporary files in the production environment. By default, the Scriptcase recommends to use: <code>/scriptcase/tmp</code> .
Documents Folder	It is necessary define the directory that will be stored the documents in the production environment. By default, the Scriptcase recommends to use: <code>/home/\$user/public_html/scriptcase/file/doc</code> . Is this option is necessary to inform the absolute path to the documents folder.
Choose the Initial Application	Defines the system initial application.

You will need to define the directories structure according to the structure of common libraries defines in your production server. Below are shown two distinct examples of settings.

Example 1:

NOTE: It is necessary to create the directories in the production server, so that it can store the deployment's **Production environment (Common Libraries)**, remembering that the absolute path until the folder of the documents directory needs to be informed. On this example, we are using the production folder.

Applications Deployment ?

Theme sc_news:news_pt

Deploy with the common files(CSS, buttons, images, messages)

- Common Libraries Folder
- Images Folder
- Temp Folder
- Documents Folder

Choose the initial application

- √ root
- √ scriptcase9nm
 - file
 - doc
 - img
 - prod
 - tmp

Example 2:

NOTE: On this example, it is necessary to create the image, documents and temporary directories, on your production server, remembering that the documents directory needs to be informed the absolute path. Relating to the **Production Environment (Common Libraries)**, it is necessary to take the "prod" folder generated at the end of the deploy to the server. In this case, we are using different directories from the previous example.

On both examples above, if you wish that after concluding the deployment process will be available to download the production environment files(Common Libraries), please mark the option **Deploy with the common files(CSS, buttons, images, messages)** .

Common Libraries Folder	Library and service package used by the deployed applications. Inform the relative path to the prod directory. Ex: <code>/production/prod</code> or <code>/prod</code> .
Images Folder (used for uploading service)	Folder where will be stored all the images, when an application uses a field of the type "Image (File Name)". Inform the relative path to the image directory: <code>/production/file/img</code> .
Temp Folder	Folder where will be stored the generated temporary files during applications' runtime. The files of this directory will be removed periodically according to the specific settings, by default it is 120 minutes. Inform here the relative path to the tmp directory. Ex: <code>/production/tmp</code> or <code>/temporay_files/tmp</code> .
Documents Folder (used for uploading serviced)	Folder where all the documents will be stored when an application uses a field of the type "Image(File Name)". Absolute Path (Full Path) to the doc directory. Ex: <code>C:/Apache/htdocs/productions/file/doc</code> or <code>C:/Apache/htdocs/my_documents/doc</code> .

NOTE: The character `/` , which means that is the root server. This character is most used to define relative paths.

Defining The Production Connection Name

Define the connection name that will be used by the project after deploying.

Connection names In this option, we will inform the names of the connections exists in your project for deployment.

Generated Files

Choose the the way method of deploy that is going to be used. ScriptCase offers 4 options:

Generate ZIP or Tar.Gz with applications

The Scriptcase will generate a file **Zip** or **Tar.Gz**. In the end of this process, will be displayed a report of the deployed applications with the download link for download the project.

NOTE: The file **ZIP** or **Tar.Gz** generated, must be decompressed only in the deployment server, to avoid that the files be corrupted.

Deploy on a server directory

Deploys the project in the informed directory. This option should be used, in the case that the deploy is going to be in the same sever where ScriptCase is installed, otherwise this option is not valid.

NOTE: This option should be used in the case that the deploy is going to be in the same sever where ScriptCase is installed, otherwise this option is not valid.

Deploy on FTP server

This option generates the files directly in the FTP remote server.

FTP server	Server where you will deploy your project. Inform the domain or the server's IP.
User	User for the FTP server in the production, remembering that the user need to have permission to write in the server folder, otherwise it will not work.
Password	Password for the FTP user.
FTP Folder	Server directory where the project will be deployed in.

NOTE: To use this option is necessary that you have a server that provides the FTP access and an user with write permissions.

Deploy on a SFTP server

Deploys the application automatically on a remote SFTP server.

SFTP server	Server where you will deploy your project. Inform the domain or the server's IP.
Port	Define the port to establish connection with SFTP server.
User	User for the SFTP server in the production, remembering that the user need to have permission to write in the server folder, otherwise it will not work.
Password	Password for the SFTP user.
SFTP Folder	Server directory where the project will be deployed in.

NOTE: To use the **SFTP** option is necessary that the user have permission to write on the server folder.

Server Directories

At the end of the process you will receive a report with all the applications deployed and a download link of the project, also with the common libraries, that are the necessary routine to run the applications on the production environment, that is why they are required to upload with the deployment.

Applications Deployment

100%

Deployment finished

Applications	Result	Processing Time
news_categorias_cad_cons	OK	0.13
news_detalhes_noticia	OK	0.09
news_editorial_cons	OK	0.10
news_editorial_list_categ_cons	OK	0.11
news_editorial_list_categ_cons_new	OK	0.11
news_editorial_por_sessao_cons	OK	0.15
news_noticias_caixa_saida_cons	OK	0.16
news_noticias_editar_publicadas	OK	0.20
news_noticias_publicar_cons	OK	0.37
news_usuarios_cad_cons	OK	0.19
apl_inicial	OK	0.16
news_troca_senha_controle	OK	0.16
login	OK	0.09
news_categorias_cad_frm	OK	0.20
news_config_sis_frm	OK	0.35
news_noticias_publicar_frm	OK	0.22
news_noticias_redigir_frm	OK	0.19
news_usuarios_cad_frm	OK	0.18
menu	OK	0.04
Processing time		3.2

SFTP!
 Applications sent successfully to SFTP server

The advanced deploy requires that you download the "Production Environment", that is the package that contains the common libraries.

NOTE: If you do not want to download the **Production Environment (Common Libraries)** from the link generated by the deploy, you can download it by [Clicking Here](#). If you deploy the project again, you do not need to download the common libraries, but this information is not valid if the common libraries were updated in a new ScriptCase release.

In the production environment we will configure the connections used in the deployed projects.

First Access

When realize the first access to production environment, for security, we must create a new password.

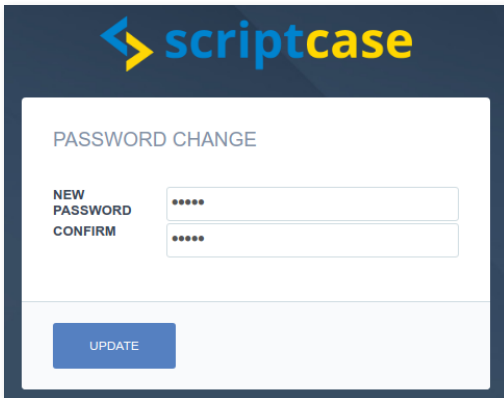
In first access, the default password always will be "scriptcase".

The production default path in a typical deployment is: `/project_name/_lib/prod` . Access Example: `http://192.168.254.196:8090/project/_lib/prod`



Accessing the production Environment

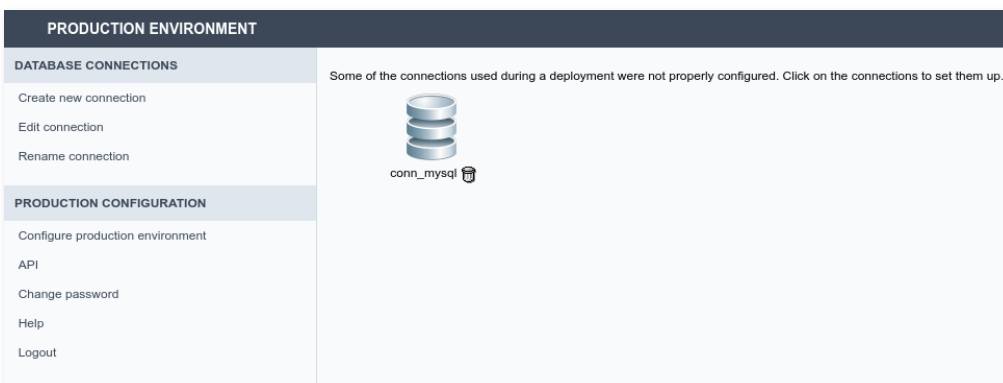
When we put the default password "scriptcase", we must define our new password.



Password configuration for production environment

Then we will be redirect to screen that inform the connections name that must be created.

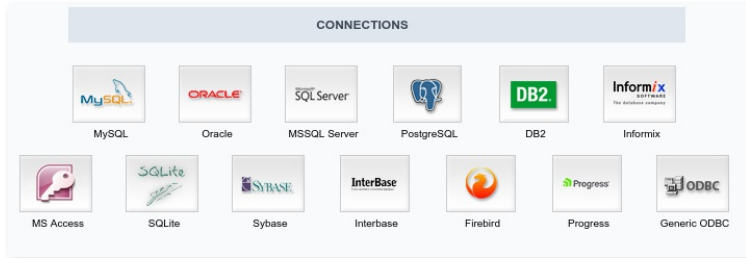
The connections must be created with the same connections names listed here, because it is about the connection names that the applications are using.



Production Environment setting screen

Create New Connection

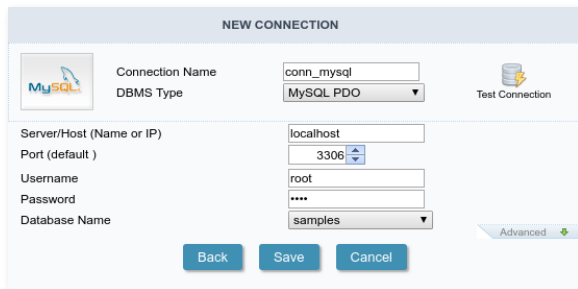
We must choose the SGBD, the database that we will use to this project.



Selecting connection in production environment

Then we must inform the Database Server. If the database is in the same server that the applications we will be able to call they trough 'localhost'. If the database is in another server, you will be able to connect the database trough the machine Name or IP adress. We will inform the Login and Password to access the database(if possess).

The connection in production environment should possess the same name in the used connection by the project in production environment.



Creating connection in production environment

- **onnection name**: Defines the name for the connection created.
- **BMS Type**: Select the Driver type to the connection.
- **erver/Host (Name or IP)**: Allows you to inform the machine Name or IP adress to your database server.
- **ort (default)**: Allows you to select a port to establish the connection.
- **ername**: Inform the username to connect with your database.
- **assword**: Inform the password to connect with your database.
- **atabase name**: Inform the database that will be used in the connection.

Rename Connection

Allows you to rename the connections in production environment.

Select the connection that you wish to rename, and inform a new name to this connection.



Renaming the connection

*This procedure **do not change the connection name that the project will use** . The connection in production environment must possess the same name of the used connection by the project in the development environment.*

API

We must configure the APIs that we are using in our project with the same informations used in the development environment.

[Click here](#) to see the APIs documentation.

#	NAME	GATEWAY	ACTION
1	PAYPAL	paypal_express	Edit Remove
2	SMS	clickatell	Edit Remove
3	SMTP	smtp	Edit Remove

What are macros?

Macros are functions created by Scriptcase developers to help you customize your application and coding. Using Macros simplifies your code editing as well as your validation processes.

The code editor within Scriptcase provides you some functionalities which allow you to change the code editor to wide screen or to display the sidebars menus with some options.

Once you're inside the the event code editor, you will see the right side bar menu with the available macros being displayed with a green point

Scriptcase's macros always starts you can type "swith a "sc_". A hint is thatc_" and hit ctrl+spacebar in your keyboard to check a list of available macros in the event you are.

Where can they be used?

You can use the macros within the application events, depending on the scope described in the article [Events X Macros](#) and within [internal](#) or [external libraries](#) from Scriptcase. application methods and fields lookups.

SCRIPTCASE MACROS

Scriptcase has macros and special variables that allow the user to manipulate events, buttons, applications, security control, manipulate with dates, etc.

Follows a table of all macros and special variables from **ScriptCase**, including the list of applications that works for each one.

MACROS AND SPECIAL VARIABLES	
SQL	
sc_begin_trans ("Connection")	This macro starts a set of transactions in the database.
sc_change_connection ("Old_Connection", "New_Connection")	This macro dynamically change the application connections.
sc_commit_trans ("Connection")	This macro effective a set of transactions in the database.
sc_concat	With this macro you can concatenate fields on select for every database.
sc_connection_edit ("Connetion_Name", \$arr_conn)	This macro edits an existing connection at runtime.
sc_connection_new ("Connection_Name", \$arr_conn)	This macro creates new connections dynamically.
sc_error_continue ("Event")	This macro deactivates the Scriptcase standard database error treatment message for an event.
sc_error_delete	This macro configure the variable that contains the database error message that can occurs during the exclusion of a record.
sc_error_insert	This macro configure the variable that contains the database error message that can occurs during the addition of a record.
sc_error_update	This macro configure the variable that contains the database error message that can occurs during the update of a record.
sc_exec_sql ("SQL Command", "Connection")	This macro execute SQL commands passed as parameter or a SQL command in the SQL field action type.
sc_lookup (Dataset, "SQL Command", "Connection")	This macro executes a SELECT command stored in the second parameter and returns the data in a variable.
sc_reset_change_connection	This macro erases the changes made using "sc_change_connection".
sc_reset_connection_edit	This macro undoes the connection edits made by macro "sc_connection_edit".
sc_reset_connection_new	This macro undoes the connections made by the macro "sc_connection_new".
sc_rollback_trans ("Connection")	This macro discards a set of transations in the data base.
sc_select (dataset, "SQL Command", "Connection")	This macro executes the commands passed in the second parameter and returns the dataset in a variable.
sc_select_field ({Field})	This macro modify dynamically a field that will be recovered in the grid.
sc_select_order ("Field")	This macro modify dynamically the grids "ORDER BY" clause field.
sc_select_where (add)	This macro adds dynamically a condition to the grid WHERE clause.
sc_set_fetchmode (parm);	This macro allows to change the type of return from the dataset of the select commands
sc_sql_injection ({My_Field}) or (\$My_Variable)	This macro is used protect the field/variable against "SQL injection".
sc_sql_protect (Value, "Type", "Connection")	This macro protects the value passed as parameter according with the used database.
sc_where_current	This macro is used to make a reference of the where clause currently used.
sc_where_orig	This macro saves the where clause content of the original application select.
Variables - Database Database Variables	Global variables containing database access values used by main connection.
Date	
sc_date (Date, "Format", "Operator", D, M, Y)	This macro calculates and returns increments and decrements using dates.
sc_date_conv ({Field_Date}, "Input_Format", "Output_Format")	This macro converts the date field passed as parameter with an input format to another field with an output format.
sc_date_dif ({Date1}, "Format Date1", {Date2}, "Format Date2")	This macro calculates the difference between two dates (passed as parameters) returning the result in days.
sc_date_dif_2 ({Date1}, "Format Date1", {Date2}, "Format Date2", Option)	This macro calculates the difference between two dates returning the amount of days, months and years.
sc_date_empty ({Field_Date})	This macro checks if a date field its empty retuning a boolean.
sc_time_dif ({datetime_01}, "Date_01 Format", {datetime_02}, "Date_02 Format")	Calculate difference in hours, returning the amount of hours, minutes and seconds.
Control	
sc_ajax_javascript ('JavascriptMethodName', array("parameter"))	This macro allows the execution of JavaScript methods in form/control events
sc_alert ("Message", \$array)	This macro shows a Javascript alert message screen.
sc_apl_conf ("Application", "Property", "Value")	This macro modify the application execution property.
sc_apl_default ('application', 'type');	This macro allows that the user defines in your initial application what will happen when the application lost the session.
sc_calc_dv (Digit, Rest, Value, Module, Weights, Type)	This macro calculate verify digits.
sc_call_api (\$profile, \$arr_settings)	This macro allows us to use as integrated Scriptcase APIs
sc_changed ({Field_Name})	This macro returns "true" if the field name have been changed.
sc_confirm ("Message")	This macro shows a Javascript confirm screen.
sc_decode ({My_Field})	This macro returns the encrypted field or variable to its original value.
sc_encode ({My_Field})	This macro returns the field or variable with the content encrypted.
sc_error_exit (URL / My_Application, "Target");	This macro Interrupts the application execution if there are error messages generated by the macro "sc_error_message".
sc_error_message ("Text")	This macro generate an error message.
sc_exit (Option)	This macro forces the application exit.
sc_getfield ('myField')	This macro assign the properties of a field to a javascript variable.
sc_get_language	This macro returns the abbreviation of the language used.
sc_get_regional	This macro returns the abbreviation of the regional settings used.
sc_get_theme	This macro returns the application theme name.
sc_groupby_label ("My_Field")	This macro dynamically modify the field label displayed in group by lines.

sc_image (Image01.jpg)	This macro loads images passed as parameter to use in the application.
sc_include ("File", "Source")	This macro is used to "include" PHP routines.
sc_include_lib ("Lib1", "Lib2", ...)	This macro is used to select dynamically the application libraries.
sc_include_library ("Target", "Library Name", "File", "include_once", "Require")	This macro includes a PHP file from a library in the application.
sc_label ({My_Field})	This macro its used to modify dynamically the grid form field label.
sc_language	This macro returns the language and regional settings.
sc_link (Column, Application, Parameters, "Hint", "Target", Height, Width)	This macro dynamically creates or modifies links between grid applications and other applications.
sc_log_add ("action", "description")	This macro will add a register into the log table.
sc_log_split ({description})	This macro returns what was inserted in the "description" field of the log table in an array format.
sc_mail_send (SMTP, Usr, Pw, From, To, Subject, Message, Mens_Type, Copies, Copies_Type, Port, Connection_Type, Attachment, SSL)	This macro is used to send e-mails.
sc_make_link (Application, Parameters)	This macro is used to create a string with the link data to another application.
sc_master_value ('Object', Value)	This macro update a Master Application object from a Detail Application.
sc_redir (Application, Parameter01; Parameter02; Target, Error, height_modal, width_modal)	This macro its used to redirects to other application or URL.
sc_reset_apl_default	This macro can reset sc_apl_default macro settings.
sc_reset_global ([Global_Variable1], [Global_Variable2] ...)	This macro delete session variables received as parameter.
sc_send_mail_api (\$arr_settings)	Enables dynamic sending of embedded emails with Mandrill and Amazon SES
sc_send_sms (\$arr_settings)	This macro is used to send text SMS messages
sc_send_whatsapp (\$arr_settings)	This macro allows the dynamic sent of text messages through Whatsapp.
sc_seq_register	This macro provide the register sequential number.
sc_set_global (\$variable_01) or ({My_Field})	This macro is used to register session variables.
sc_set_groupby_rule	Macro is used to select an specific GROUP BY rule.
sc_set_language ('String Language')	This macro allows to dynamically change the application language.
sc_set_regional ('String Regional')	This macros allows to dynamically change the application regional settings.
sc_set_theme ('String Theme')	This macro is used to dynamically define the application themes.
sc_site_ssl	This macro verifies if its been used a safe/secure site. (https protocol)
sc_trunc_num ({My_Field}, Decimal_Number)	This macro its used to set the number of decimals.
sc_url_exit (URL)	This macro modifies the application exit URL.
sc_url_library ("Target", "Library Name", "File")	This macro returns the path of a file, inside a library, to be used on the applications.
sc_warning 'on' or 'off'	This macro dynamically activates or deactivates warning messages control.
sc_webservice ("Method", "URL", "Port", "Send Method", "Parameters Array", "Setting's Array", "Timeout", "Return")	This macro is used to communicate with a web service.
sc_zip_file ("File", "Zip")	This macro its used to generate ZIP files from a file list and/or directories.
Variables - Totalling Totalling Variables	Variables that contains all the totals (general and for grouping).
Variables - Totalling (group by) Totalling Variables (group by)	Breaking totals variables.
Filter	
sc_where_filter	This macro its used to save the where clause content generated through the filter form.
Security	
sc_apl_status ("Application", "Status")	This macro Activate/Deactivate the applications at user level.
sc_ldap_login	This macro establish the connection with the user credentials.
sc_ldap_logout ()	Macro used to release the connection after using the macro sc_ldap_login
sc_ldap_search (\$filter = 'all', \$attributes = array())	Macro to perform searches in the LDAP.
sc_reset_apl_conf ("Application", "Property")	This macro deletes all the modifications effected by "sc_apl_conf" macro.
sc_reset_apl_status	This macro deletes all the application security status variables.
sc_reset_menu_delete	This macro restores a menu item structure. (removed by the macro "sc_menu_delete").
sc_reset_menu_disable	This macro its used to enable a menu item structure. (disabled by the macro "sc_menu_disable").
sc_user_logout ('variable_name', 'variable_content', 'apl_redir.php', 'target')	Macro used to log the user out to the system.
Shows	
sc_ajax_message ("Message", "Title", "Parameters", "Parameters_Redir", "String_toast");	This macro allows the application to display customized messages.
sc_ajax_refresh	Macro to refresh a Grid
sc_block_display (Block_Name, on/off)	This macro dynamically show/hide the fields of a specific block.
sc_field_color ("Field", "Color")	This macro changes the color of a determined field text.
sc_field_disabled ("Field_Name = True/False", "Parameter")	This macro is used to block a field to get any data that would be typed on it.
sc_field_disabled_record ("Field_Name = True/False", "Parametre")	This macro has the objective to block the typing on determined fields in the Forms.
sc_field_display ({My_Field}, on/off)	This macro dynamically display or not a specific field.
sc_field_init_off (Field1, Field2,...)	This macro is intended to inhibit the query fields on the initial load.
sc_field_readonly ({Field}, on/off)	This macro dynamically set a form field attribute to ""Read-Only"
sc_field_style ({My_Field}, "Background-Color", "Size", "Color", "Family", "Weight")	This macro allows to modify dynamically the grid field style.
sc_format_num ({My_Field}, "Group_Symb", "Dec_Symb", "Amount_Dec", "Fill_Zeros", "Side_Neg", "Currency_Symb", "Side_Currency_Symb")	This macro its used to format numerical values.
sc_format_num_region ({My_Field}, "Qtde_Dec", "Insert_Zeros", "Monetary_Sym")	This macro has the objective to format numbers, using the regional settings
sc_form_show 'on' or 'off'	This macro dynamically show or not the form.
sc_get_groupby_rule ()	This macro provides the name of the Group By rule running at the time.
sc_hide_groupby_rule ('group1', 'grop2', 'group3')	Macro used to disable Group By rules.
sc_set_focus ('Field')	This macro its used to set the focus to a form field.
sc_text_style ({My_Field}, "Background-Color", "Size", "Color", "Family", "Weight")	This macro allows to modify dynamically the text style from the grid field.

Buttons	
sc_btn_copy	This macro returns "true" when the "copy" button is selected in a form.
sc_btn_delete	This macro returns "true" when the "Delete" button is selected in a form.
sc_btn_display ("Button_Name", "on/off")	This macro shows and hides buttons on the toolbar in execution time.
sc_btn_insert	This macro returns "true" when the "Add" button is selected in a form.
sc_btn_new	This macro returns "true" when the "Add New" button is selected in a form.
sc_btn_update	This macro returns "true" when the "Save" button is selected in a form.
PDF	
sc_set_pdf_name "sc_my_file_name.pdf"	This macro will change the grid's exported files name.
Variables - Authentication Authentication Variables	User/Password for the WEB server.
Menu	
sc_appmenu_add_item ("Menu_Name", "Id_Item", "Id_Parent", "Label", "Application", "Parameters", "Icon", "Hint", "Target")	This Macro adds dinamically an item to the menu.
sc_appmenu_create ("Menu_Name")	This macro dynamically creates a menu item.
sc_appmenu_exist_item ("Menu_Name", "Id_Item")	This macro checks if there is a menu item.
sc_appmenu_remove_item ("Menu_Name", "Id_Item")	This macro removes dynamically a menu item.
sc_appmenu_reset ("Menu_Name")	This macro reset the array used in the dinamically creation of a menu application.
sc_appmenu_update_item ("Menu_Name", "Id_Item", "Id_Parent", "Label", "Application", "Parameters", "Icon", "Hint", "Target")	This macro updates a menu item.
sc_btn_disable ('button_id', 'on/off')	Macro used to disable Menu buttons.
sc_menu_delete (Id_Item1)	This macro remove items of the menu structure.
sc_menu_disable (Id_Item1)	This macro deactivate menu structure items.
sc_menu_force_mobile (boolean)	Macro used to force the creation of menus to mobile devices.
sc_menu_item	This macro Identifies the menu item selected.
sc_script_name	This macro identifies the application name that was selected in the menu.

```
sc_ajax_javascript( 'JavascriptMethodName', array("parameter"))
```

This macro allows the execution of JavaScript methods called from some events in **inform, control and calendar applications**.

To use this macro you need to create a javascript method in the [Programming settings](#).

This macro receive two parameters:

The first parameter (required) receive the javascript method name.
The second (optional) receive an array with the method parameters on it.

Examples:

Ex. 1: Method javascript without parameters:

```
sc_ajax_javascript('method_name');
sc_ajax_javascript('sum');
```

Ex. 2: Method javascript with a parameter:

```
sc_ajax_javascript('method_name', array("parameter"));
```

Ex. 3: Method javascript with two parameters:

```
sc_ajax_javascript('method_name', array("param1","param2"));
```

Example using global variables:

```
sc_ajax_javascript('sum', array([num1],[num2]));
```

Macro Scope

calendar application	Form application	Control Form
ajaxFieldonBlur	ajaxFieldonBlur	
Onchange	Onchange	
OnClick	OnClick	
OnFocus	OnFocus	
onAfterDelete	onAfterDelete	onBlur
onAfterInsert	onAfterDeleteAll	onChange
onAfterUpdate	onAfterInsert	onClick
onValidate	onAfterInsertAll	onFocus
onValidateFailure	onAfterUpdate	onValidateFailure
onValidateSuccess	onAfterUpdateAll	
	onValidate	
	onValidateFailure	
	onValidateSuccess	

```
sc_ajax_message("Message", "Title", "Parameters", "Parameters_Redir", "String_toast");
```

This macro allows the application to show customizable messages. Its use is restricted to **AJAX events of Form/Control applications**.

This macro is only available in applications with **AJAX support**.

Parameters	Description
Message	The message to be shown. This parameter is required.
Title	Messagebox title, if this parameter is empty, the title will not be shown.
Message Parameters String	Message format parameters. Ex.: modal=Y&button=Y&button_label=Ok&redir=form_employees.php&redir_target=_blank For more details about this parameter, see Table2.
Redirecting Parameters String	In this section it is possible to define values for the redirecting target application. This parameter works only if a "redir" property was informed. The properties for this parameter must be informed in this format: parameter=value;parameter=value. Ex.: customer_id=ALFKI;customer_name=ALBERT FRANCINET
Toast parameter string	Message formatting parameter using Toast. - toast : Y ou N - toast_pos : 'top', 'top-start', 'top-end', 'center', 'center-start', 'center-end', 'bottom', 'bottom-start', 'bottom-end' - type : 'warning', 'error', 'success', 'info', 'question'

Table2: Message Parameters String

Property	Description	Values	Example
modal	Flag to define if the message will be shown in modal	Y or N	modal=Y
timeout	Time in seconds to hide the message. If this property is not informed, the message will be shown indefinitely. This property works only if "button" property is not enabled.	Seconds	timeout=5
button	Shows a button inside the message. This button default function is to close the message. If there is redirecting enabled, the application will be redirected after closing the message.	Y or N	button=Y
button_label	Button label. This option works only if the "button" property is enabled.	String	button=Y&button_label=OK
top	Message positioning value in pixels from	Pixels Number	top=300

left	top margin. Message positioning value in pixels from left margin.	Pixels Number	left=200
width	Messagebox width value in pixels	Pixels Number	width=320
height	Messagebox height value in pixels	Pixels Number	height=30
redir	URL or application name for redirecting. This macro works only if the "button" property is enabled.	String	button=Y&redir=form_employees.php
redir_target	Target for redirecting. This property works only when the "redir" property is enabled.	String	redir=form_employees.php&redir_target=_blank
show_close	Flag for exhibition of the "close" button on the messagebox title bar. When activated, this parameter forces a title bar to be shown, even if there is no title defined.	Y or N	show_close=Y
body_icon	Flag for exhibiting a icon on the message body. The icon exhibition depends on the applications theme configuration (exhibition schemes).	Y or N	body_icon=N

Example 01

```
sc_ajax_message ("Message", "Title", "timeout=0");
```

Example 02: Using Toast

```
sc_ajax_message ("Client", "Title", "toast=Y&toast_pos=center-start&type=success");
```

To display the message using Toast, this option must be enabled in the application interface.

Macro Scope

calendar application	Grid application	Form application	Control Form
ajaxFieldonBlur Onchange OnClick OnFocus	onClick	ajaxFieldonBlur Onchange OnClick OnFocus	onBlur onChange onClick onFocus

sc_ajax_refresh

This macro is used especially in Ajax events of a Grid with the purpose of reload the Grid data

(refresh).

Ex:

```
sc_ajax_refresh();
```

Macro Scope

Grid application

onClick

sc_alert("Message", \$array)

This macro sends a message in the alert style of Javascript (Sweetalert) or Sweetalert2.

Ps

To display the message using Sweetalert2, this option must be enabled in the application interface.

Ex. 1: Mensagem utilizando o Sweetalert

```
sc_alert("This is an alert message!");
```

Ex. 2: Mensagem utilizando o Sweetalert2

```
$params = array(
'title' => 'Title',
'type' => 'success',
'timer' => '2000',
'showConfirmButton' => false,
'position' => 'bottom-end',
'toast' => true
);
```

```
sc_alert("Inserted successfully!", $params);
```

Macro Scope

Blank application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu
onExecute	onApplicationInit onFooter onHeader onScriptInit	onApplicationInit onFooter onHeader onScriptInit onNavigate onRecord	onApplicationInit onFilterInit	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad

sc_apl_conf("Application", "Property", "Value")

This macro allows to modify the property of the application. The user can determine if a Form will start in insert mode, or a Grid starts by filter, etc.

This macro **must** be used by one application to modify the properties of another application.

Properties for Form applications

Property	Value	Description
start	new	Forces the form to start in the insert mode in order to add new records.
insert	on/off	Qualifies (on) or unqualifies (off) the "add new" button to allow the addition of new records.
update	on/off	Qualifies (on) or unqualifies (off) the "save" button on the form.
delete	on/off	Qualifies (on) or unqualifies (off) the "delete" button on the form.
field_display_off	field	Dinamically hides a field.
field_display_on	field	Dinamically shows a field.
field_readonly	field	Dinamically set the "readonly" attribute in order to select the field.
rows		Forces the number of lines for each page (only for form of multiple records).
rows_ins		Forces the number of lines for insert (only for form of multiple records).

Property for Grid applications

Property	Value	Description
start	filter	Forces the Grid to start by the filter.
cols		Forces the number of columns. (vertical and slide kind)
rows		Forces the number of lines per page.
lig_edit	on/off	Qualifies (on) or unqualifies (off) the record edition. (the pencil icon)

Property for all applications

Property	Value	Description
exit	apl/url	Forces application to exit to a specified location.

Ex. 1: Forces the "my_form" application to start in the addition mode.

```
sc_apl_conf("my_form", "start", "new");
```

Ex. 2: Doesn't allow the "my_form" application make addition of new registers.

```
sc_apl_conf("my_form", "add", "off");
```

Ex. 3: Sets "my_field" on the application "my_form" (readonly attribute) to "true" dinamically.

```
sc_apl_conf("my_form", "field_display_off", "my_field");
```

Ex. 4: Shows "my_field" on the application "my_form" dinamically.

```
sc_apl_conf("my_form", "field_display_on", "my_field");
```

Ex. 5: Forces the "my_grid" application to start by filter.

```
sc_apl_conf("my_grid", "start", "filter");
```

Ex. 6: Forces the "my_grid" application to show 20 lines.

```
sc_apl_conf("my_grid", "rows", "20");
```

Ex. 7: Hides "my_field" on the application "my_form" dinamically

```
sc_apl_conf("my_form", "field_display_off", "my_field");
```

Ex. 8: Use the button on the toolbar menu.

```
if ({sc_menu_item} == "btn_1")
{
sc_apl_conf("form_customer", "start", "new");
}
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	onApplicationInit onScriptInit	onApplicationInit onScriptInit	onApplicationInit onScriptInit	onApplicationInit onFilterInit	onApplicationInit onScriptInit	onApplicationInit onScriptInit onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onApplicationInit onScriptInit

sc_apl_default('application', 'type');

This macro allows that the user defines in your initial application what will happen when the application lost the session.

The macro has the following parameters:

Apl: Here it is necessary to inform which application the macro will redirect after lost the session.

Type: Here we will inform the redirecting type that will occur. There are two types:

- **R:** The application will only redirect to the initial application.
- **M:** Will be shown a message before the redirecting.

Example: In a login application defined as the initial application, we can define that this application will redirect to himself.

```
sc_apl_default("app_login", "R")
```

Macro Scope

calendar application	chart application	Grid application	Search application	Form application	Control Form	ReportPDF application
onCalendarScriptInit onScriptInit	onScriptInit	onScriptInit	onFilterInit	onScriptInit	onScriptInit	onScriptInit

sc_apl_status("Application", "Status")

Used to control the security access. These macros activate/deactivate the applications that a user has access to.

Application: Is the name or the variable that contains the name of the application to be activated/deactivated.

Status: Is the value or variable that contains the value to be attributed to the application. The values are: "on" to activate and "off" to deactivate.

Ex. 1:

sc_apl_status ('atu_cad', 'off');

Ex. 2:
 sc_apl_status ({var_name_apl}, {var_status});

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	onApplicationInit onCalendarApplicationInit onCalendarScriptInit onScriptInit	onApplicationInit onScriptInit	onApplicationInit onScriptInit	onApplicationInit onFilterInit	onApplicationInit onScriptInit	onApplicationInit onScriptInit onValidate onValidateFailure onValidateSuccess	onApplicationInit onLoad	onApplicationInit onLoad	onApplicationInit onScriptInit

sc_appmenu_add_item("Menu_Name", "Id_Item", "Id_Parent", "Label", "Aplication", "Parameters", "Icon", "Hint", "Target")

Adds dinamically an item to a menu application.

It is only possible to use this macro when there is an array created before by the macro *sc_appmenu_create*.

Parameter	Description
Menu_name	Application menu name.
Id_item	Menu item id.
Id_Parent	Parent menu item id. For roots the null/empty value is passed.
Label	Menu item description.
Aplication	Menu item URL or application name
Parameters	String with parameters to pass to the menu item application. Ex.: param1=valor;param2=valor
Icon	Path to icon used in menu item.
Hint	Descriptive hint to menu item.
Target	Target for menu item link. ("_self" to use the same window, "_blank" to a new one and "_parent" to exit the window.

Ex. 1: Creates a menu named menu_main.
 sc_appmenu_create('menu_main');

Ex. 2: Adds a on the menu_main the the item_1 named Category.
 sc_appmenu_add_item('menu_main','item_1','','Category');

Ex. 3: Adds a on the menu_main (item_2) the item_1 name Category linked to the form_category passing a few parameters.
 sc_appmenu_add_item('menu_main','item_2','item_1','Category Form','form_category','param1=value;param2=value');

Macro Scope

Menu application	Tree menu
onApplicationInit onLoad	onApplicationInit onLoad

sc_appmenu_create("Menu_Name")

This macro starts or array for dynamic montagem of a menu application. You must be informed or you will not apply the menu as a parameter.

This macro should be used in events of control or not **event onload of menu**.

Ex. 1:
 sc_appmenu_create ('menu_main');

Macro Scope

Menu application	Tree menu
onApplicationInit onLoad	onApplicationInit onLoad

sc_appmenu_exist_item("Menu_Name", "Id_Item")

Check if a menu item exists and return a true or false value.

Parameter	Description

Menu Name Id_Item	Menu App Name Menu Item Identifier Code
----------------------	--

Ex. 1:

```
if(sc_appmenu_exist_item('menu_main', 'item_2')){
sc_appmenu_update_item('menu_main','item_2','item_1','Category Form','form_category');
}
else{
sc_appmenu_add_item('menu_main','item_2','item_1','Category Form','form_category');
}
}
```

Macro Scope

Menu application	Tree menu
onApplicationInit	onApplicationInit
onLoad	onLoad

sc_appmenu_remove_item("Menu_Name", "Id_Item")

Remove dynamically a menu item.

To use this macro it **is necessary to execute the macro `sc_appmenu_create` before.**

Parameter	Description
menu_name	Menu application name
id_item	Menu item id

Ex. 1:

```
sc_appmenu_remove_item('menu_main','item_2');
```

Macro Scope

Menu application	Tree menu
onApplicationInit	onApplicationInit
onLoad	onLoad

sc_appmenu_reset("Menu_Name")

This macro resets the array used in the dynamic creation of a menu application.

This macro is expecting the menu application name as the single parameter.

Ex. 1:

```
sc_appmenu_reset('menu_main');
```

Macro Scope

Menu application	Tree menu
onApplicationInit	onApplicationInit
onLoad	onLoad

sc_appmenu_update_item("Menu_Name", "Id_Item", "Id_Parent", "Label", "Application", "Parameters", "Icon", "Hint", "Target")

Updates an application menu item dynamically

It is only possible to use this macro when there is an array created before by the macro `sc_appmenu_create`.

It is only possible to use this macro on items created before by the macro `sc_appmenu_add_item`.

Parameter	Description
menu_name	Application menu item name
id_item	Application menu item id
id_parent	Application menu item parent id. For roots the null/empty value is passed.
Label	Application menu item description.
Application	Menu item URL or application name
Parameters	String with parameter to pass to the application menu item Ex.: param1=valor;param2=valor

Icon	Icone path used in menu item.
Hint	Descriptive Hint for menu item.
Target	Target for menu item link

Ex. 1:
`sc_appmenu_update_item('menu_main','item_2','item_1','Category Form','form_category');`

Macro Scope

Menu application	Tree menu
onApplicationInit	onApplicationInit
onLoad	onLoad

sc_begin_trans("Connection")

This macro starts a set of transactions in the database.

For the use of this macro you need to have a **transactional** database.

In form applications, the events (onAfterInsert, onAfterUpdate, onAfterDelete, onBeforeInsert, onBeforeUpdate or onBeforeDelete) are automatically protected by a transaction control, since the connection is the same of the application.

In the other cases, if the user need to make a transaction control, must begin with this macro and finish with the [sc_commit_trans\(\)](#) macro to confirm the updates or [sc_rollback_trans\(\)](#) to cancel the transactions.

The "connection" parameter is optional, necessary only if the command is executed in a different connection from the actual application.

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onFooter onHeader onScriptInit	onClick onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onFooter onHeader onScriptInit onRecord

sc_block_display(Block_Name, on/off)

Dynamically determines the display fields of a specific block.

By default all the blocks are displayed ("on" condition).

Ex. 1:

```

if ({type_customer} == "personal")
{
sc_block_display(company, off);
}
else
{
sc_block_display(personal, off);
}

```

Obs: In grids, this macro only works with "slide" orientation.

Macro Scope

calendar application	chart application	Grid application	Form application	Control Form
onAfterInsert onAfterUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onScriptInit	onScriptInit onRecord	onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onClick onScriptInit onLoadAll onRefresh

sc_btn_copy

Available when the "Copy" button is clicked. Can be tested and used inside the ScriptCase events, allowing specific programming in run time.

Ex. 1:

```
if (sc_btn_copy)
{
sc_message("Record copied successfully!");
}
```

Macro Scope

calendar application	Form application
onScriptInit onLoad	onScriptInit onLoad

sc_btn_delete

Available when the "Delete" button is clicked. Can be tested and used inside the ScriptCase events, allowing specific programming in run time.

Ex. 1:

```
if (sc_btn_delete)
{
sc_error_message("Unable to delete this record");
}
```

Macro Scope

calendar application	Form application
onValidate onValidateFailure onValidateSuccess	onValidate onValidateFailure onValidateSuccess

sc_btn_disable('button_id', 'on/off')

This macros is used to disable or enable buttons from a Menu application.

Ex:

```
sc_btn_disable('btn_2', 'off');
```

Macro Scope

Menu application	Tree menu
onApplicationInit onLoad	onLoad

sc_btn_display("Button_Name", "on/off")

Activate toolbars buttons on the application in execution time.

Button_Name: Is the variable name that contains the name of the button to be activated/deactivate.

Buttons Name (Grid):

groupbyDisplays the "Group By" button in the grid.

Buttons Name	Description
first	Goes to the first page of the grid.
back	Goes to the previous page of the grid.
forward	Goes to the next page of the grid.
last	Goes to the last page of the grid.
filter	Displays the "Filter" button.
pdf	Generates colored PDF.
det_pdf	Generates detail PDF.
pdfbw	Generates black and white PDF.
xls	Generates the XLS.
xml	Generates the XML.
cvs	Generates the CVS.
rtf	Generates the RTF.
word	Generates the Word.
print	Displays the "Print" button in the grid.
det_print	Displays the "Print" Button in the detail.
summary	Displays the "Summary" button in the grid.
new	Displays the "New" button. (if the application has link for it, or if the form is running in the grid iframe)
insert	Displays the "Include" button of the form. (only for forms running in grid's iframe)
update	Displays the "Update" button of the form. (only for forms running in grid's iframe)
delete	Displays the "Delete" button of the form (only for forms running in grid's iframe)
qsearch	Displays the "Quick Search" button in the grid.
dynsearch	Displays the "Dynamic Search" button in the grid.
gridsave	Displays the "Save Grid" button in the grid.

Buttons Name (Form):

Buttons Name	Description
first	Goes to the first page of the form.
back	Goes to the previous page of the form.
forward	Goes to the next page of the form.
last	Goes to the last page of the form.
new	Displays the "New" button on the form, if its off the form will starts in insertion mode.
insert	Displays the "Include" button of the form.
update	Displays the "Update" button of the form.
delete	Displays the "Delete" button of the form.
copy	Displays the "Copy" button of the form.
qsearch	Displays the "Quick Search" button of the form.
pdf	Generates PDF file from form.
print	Open the form in print mode.
dynsearch	Displays the "Dynamic Search" button in the grid.

Buttons Name (Control):

Buttons Name	Description
ok	Displays the "Ok" button on a control.
exit	Displays the "Exit" button on a control.
facebook	Displays the "Facebook" button on a control.
twitter	Displays the "Twitter" button on a control.
google	Displays the "Google+" button on a control.
paypal	Displays the "PayPal" button on a control.

OBS. Remember that the buttons name must be lowercase.

Ex. 1:

```
sc_btn_display ('new', 'off');
```

Ex. 2: If we use variables or fields ({Fields} or [Global_Variables]) as parameters its not allowed the use of the quotes or single quotes.

```
sc_btn_display ({variavle_button}, 'off');
```

Macro Scope

calendar application	chart application	Grid application	Form application	Control Form
onAfterDelete onAfterInsert onAfterUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onScriptInit	onScriptInit	onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onScriptInit onLoadAll onRefresh onValidate onValidateFailure

sc_btn_insert

Available when the "Add" button is clicked. Can be tested and used inside the ScriptCase events, allowing specific programming in run time.

Ex. 1:

```
if (sc_btn_insert)
{
sc_message("Record inserted successfully");
}
```

Macro Scope

calendar application	Form application
onValidate onValidateFailure onValidateSuccess	onValidate onValidateFailure onValidateSuccess

sc_btn_new

Available when the "Add New" button is clicked. Can be tested and used inside the ScriptCase events, allowing specific programming in run time.

Ex. 1:

```
if (sc_btn_new)
{
{My_Date} = date('Y/m/d');
}
```

Macro Scope

calendar application	Form application
onLoad	onLoad

sc_btn_update

Available when the "Save" button is clicked. Can be tested and used inside the ScriptCase events, allowing specific programming in run time.

Ex. 1:

```
if (sc_btn_update)
{
sc_error_message("Record updated successfully");
}
```

Macro Scope

calendar application	Form application
onValidate onValidateFailure onValidateSuccess	onValidate onValidateFailure onValidateSuccess

sc_calc_dv(Digit, Rest, Value, Module, Weights, Type)

Calculate verifier (checksum) digits.

Parameter

Parameter	Description
Digit	Variable that will receive the calculated digit.
Rest	Variable that will receive the "rest" of the division.
Value	Value or Variable to calculate the digit.
Module	Value or Variable containing the module to be used. Default value is module "11".
Weights	Value or Variable containing the weights to be used. Default value is the "98765432" values.
Type	Value or Variable containing the parcels calculation to be used: 1 to normal sum of the parcels values and 2 to the sum of each algorithm of the parcels. Default value is "1".

Ex. 1:

```
sc_calc_dv({my_dv}, {my_rest}, 1234567, 11, 98765432, 1);
Parcels calculation: 7x2= 14 ; 6x3=18 ; 5x4=20 ; 4x5=20 ; 3x6=18 ; 2x7=14 ; 1x8=8
Parcels sum : 14 + 18 + 20 + 20 + 18 + 14 + 8 = 112
Rest of the division of 112 / 11 = 2
Digit: 11 - 2 = 9
```

Ex. 2:

```
sc_calc_dv({my_dv}, {my_rest}, 1234567, 10, 12, 2);
Parcels calculation: 7x2= 14 ; 6x1=6 ; 5x2=10 ; 4x1=4 ; 3x2=6 ; 2x1=2 ; 1x2=2
Parcels sum: 1 + 4 + 6 + 1 + 0 + 4 + 6 + 2 + 2 = 26
Rest of the division of 26 / 10 = 6
Digit: 10 - 6 = 4
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onClick onApplicationInit onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onExecute onLoad	onApplicationInit onFooter onHeader onScriptInit onRecord

sc_call_api(\$profile, \$arr_settings)

It allows us to use Scriptcase built-in APIs. We currently offer integration with the Mandrill and Amazon SES APIs.

This macro has two parameters:

Parameters	Description
\$profile	Required parameter. Can be passed empty "" or contain the gateway name configured in "tools> API".
\$arr_settings	This parameter is optional as long as the "profile" is not empty. When informed, this parameter must contain an Array with the Gateway used and the API key.

Example 1 - Parameter \$arr_settings, array with the API information

```
$settings = array( 'settings' => ['gateway' => 'mandrill', 'api_key'=> {api_key}]);
$mandrill = sc_call_api("", $settings);
```

Example 2 - Sending email using mandrill.

```
$txt_no_tags = strip_tags({msg});
$arr_merge = array(
    'name' => "",
    'type' => 'to',
    'email' => 'exemplo@exemplo.com'
);
```

```
$var_config = array( 'settings' => ['gateway' => 'mandrill', 'api_key'=> {api_key}]);
$mandrill = sc_call_api("", $var_config);
$var_msg = array(
    'from_email'      => {from_email},
    'from_name'      => {from_name},
    'html'           => {msg},
    'text'           => $txt_no_tags,
    'txt'            => array($arr_merge),
    'subject'        => {subject},
    'important'      => true,
    'auto_text'      => null,
    'auto_html'      => null,
    'inline_css'     => null,
    'metadata'       => array('website' => 'www.scriptcase.net')
);

$async = false;
$return = $mandrill->messages->send($var_msg, $async);
```

Example 3 - Sending SMS using clickatell.

```
$var_config = array(
    'message' => [
        'to' => {sms_to},
        'message' => {sms_txt},
    ],
    'settings' => [
        'gateway' => {gateway},
        'auth_token' => {auth_token},
    ]
);
```

```
sc_send_sms($var_config);
```

Example 4 - Using the macro with PagSeguro

```
$arr_settings = [
    'gateway' => 'pagseguro',
    'environment' => 'sandbox',
    'auth_email' => 'exemplo@gmail.com',
    'auth_token' => '72AE21503DDA4840BE1DC7945F6D1CE1'
];
sc_call_api("", $arr_settings);
```

Example 5 - Using the macro with PayPal

```
$arr_settings = [
    'gateway' => 'paypal',
    'environment' => 'sandbox',
    'auth_email' => 'exemplo@gmail.com',
    'auth_token' => '72AE21503DDA4840BE1DC7945F6D1CE1'
];
sc_call_api("", $arr_settings);
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	onAfterDelete onAfterInsert onAfterUpdate onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onFooter onHeader onScriptInit	onClick onHeader onScriptInit onRecord	onFilterInit onFilterSave onFilterValidate	onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onFooter onHeader onScriptInit onRecord

sc_changed({Field_Name})

Macro to check if the field name have been changed.

```
Ex. 1:
if (sc_changed({my_field}))
{
    Echo "Filed have been changed";
}
```

Macro Scope

calendar application	Form application	Control Form
onBeforeUpdate onValidate onValidateFailure onValidateSuccess	onBeforeUpdate onValidate onValidateFailure onValidateSuccess	onRefresh

sc_change_connection("Old_Connection", "New_Connection")

This macro dynamically change the application connections

Ex. 1: To change the connections:

```
sc_change_connection("Old_Concection01", "New_Connection01" ; "Old_Concection02", "New_Connection02");
```

Allows to change more then one application/connection in one command. Multiple connections can be passed as parameters that must be separated by a ";".

Ex. 2: Using variables:

```
sc_change_connection([Global_Connetion], [Test_Connection]);
```

The changes take effect in the following applications.

Ex. 3: Changing the application main connection

```
sc_change_connection({Password}, [Test] ; "My_Connection", [Global_Connection]);
```

The changes take effect in the following applications.

Ex. 4: To delete the exchange:

To recover your previous settings you can use the [sc_reset_change_connection\(\)](#) macro

Macro Scope

Blank application	calendar application	chart application	Grid application	Form application	Control Form	ReportPDF application
onExecute	onScriptInit onLoad	onHeader onScriptInit	onApplicationInit onScriptInit	onScriptInit onLoad onLoadRecord	onScriptInit onLoadAll	onHeader onScriptInit onRecord

sc_commit_trans("Connection")

This macro is used to confirm a transaction set in the database.

In form applications, there are events that perform transactions with the database (Ex: onAfterInsert, onAfterUpdate, onAfterDelete, onBeforeInsert, onBeforeUpdate, onBeforeDelete, etc...).

The transactions control of these events is done by Scriptcase itself, if the connection provided is the same of the application.

However, if the user has to perform any type of redirection in these events, it is necessary to use the macro `sc_commit_trans` before the redirect to secure the transactions previously performed.

Ex: onAfterUpdate - Using redirection after update a record:

```
sc_commit_trans();
```

```
sc_redir('grid_main');
```

The "connection" parameter is optional, being necessary only if the command is executed in a different database specified to the application.

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onFooter onHeader onScriptInit	onClick onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure	onBlur onChange onClick onFocus onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onFooter onHeader onScriptInit onRecord

sc_concat

This macro is intended to concatenate table fields in the selects.

This macro has the purpose of concatenating table fields in selects made in the events and in the application's SQL, and can be used in select fields, text-autocomplete fields and any other field that can be done select automatic.

Ex. 1: Use(Editing Lookup):

```
SELECT field1, sc_concat(field2,'-',field3) FROM table
SELECT field1, sc_concat(field2,'|',field3) FROM table
```

Ex. 2: Using the macro with [sc_lookup](#):

```
sc_lookup(Dataset,"SELECT field1,sc_concat(field2,'-',field3) FROM table");
```

Makes the formatting according to your bank.

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateSuccess	onFooter onHeader onScriptInit	onClick onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onFilterInit onFilterRefresh onFilterSave onFilterValidate	OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateSuccess	onBlur onChange onClick onFocus onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onExecute onLoad	onFooter onHeader onScriptInit onRecord

sc_confirm("Message")

Valid only in the PHP programming code that runs when called through a toolbar button.

This macro is used to confirm that the action must be really executed, preventing mistakes.

Ex. 1:

```
sc_confirm ("Do you really wish to execute this action??");
```

Macro Scope

Form application	Control Form
onClick	onClick

sc_connection_edit("Connetion_Name", \$arr_conn)

1º Parameter: Connection name to be edited.

2º Parameter: Array of items containing the connection information to be edited. Check out the indices of the array:

Indice	Description	Example
['server']	Database server (host)	\$arr_conn['server'] = "127.0.0.1"
['user']	Database username	\$arr_conn['user'] = "root"
['password']	Database password	\$arr_conn['password'] = "secretpass123"
['database']	Database name used in the connection	\$arr_conn['database'] = "sc_samples"

['persistent']	Defines if the connection is persistent or not	\$arr_conn['persistent'] = "Y" / "N"
['encoding']	Configure the connection encoding	\$arr_conn['encoding'] = "utf8"

Note: Is not required to use all the indices in the array, we can pass only the required ones.

Example:

```
$arr_conn = array();
```

```
$arr_conn['user'] = "admin2";
$arr_conn['password'] = "admin2pass";
$arr_conn['database'] = "sc_samples2";
```

```
sc_connection_edit("sc_connection", $arr_conn);
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onCalendarApplicationInit onCalendarScriptInit onScriptInit onLoad onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onScriptInit	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onExecute onLoad	onApplicationInit onFooter onHeader onScriptInit onRecord

sc_connection_new("Connection_Name", \$arr_conn)

When used, the macro creates a new connection with the information inserted in its parameters. This connection is available throughout the current session Scriptcase and cease to exist when the session is closed.

1º Parameter: Connection name.

Note: If there is a connection created within the Scriptcase with the same name, this macro has no effect. Connections created within the Scriptcase prevail. If you want to edit an existing connection, see the documentation for the macro **sc_connection_edit**.

2º Parameter: Array of items containing the connection information. Check out the indices of the array:

```
['drive']Driver of the database used for the connection (see table below)$arr_conn['drive'] = "oci8"
```

Indice	Description	Example
['server']	Database server (host)	\$arr_conn['server'] = "127.0.0.1"
['user']	Database username	\$arr_conn['user'] = "root"
['password']	Database password	\$arr_conn['password'] = "secretpass123"
['database']	Database name used in the connection	\$arr_conn['database'] = "sc_samples"
['persistent']	Defines if the connection is persistent or not	\$arr_conn['persistent'] = "Y" / "N"
['encoding']	Configure the connection encoding	\$arr_conn['encoding'] = "utf8"

Note: It is required that all items are filled, with the exception of items **['persistent']** and **['encoding']**.

See below the driver's list:

Driver	Descrição
access	MS Access
ado_access	MS Access ADO
odbc	ODBC Genérico
db2	DB2
db2_odbc	DB2 ODBC Nativo
odbc_db2	DB2 ODBC Genérico
odbc_db2v6	DB2 ODBC Genérico 6 ou Abaixo

pdo_db2_odbc	DB2 PDO ODBC
pdo_ibm	DB2 PDO
firebird	Firebird
pdo_firebird	Firebird PDO
borland_ibase	Interbase 6.5 ou Acima
ibase	Interbase
pdo_informix	Informix PDO
informix	Informix
informix72	Informix 7.2 ou Abaixo
ado_mssql	MSSQL Server ADO
pdo_sqlsrv	MSSQL Server NATIVE SRV PDO
mssqlnative	MSSQL Server NATIVE SRV
odbc_mssql	MSSQL Server ODBC
mssql	MSSQL Server
pdo_mysql	MySQL PDO
mysqlt	Mysql (Transactional)
mysql	MySQL (Non-Transactional)
pdo_oracle	Oracle PDO
oci805	Oracle 8.0.5 ou Acima
odbc_oracle	Oracle ODBC
oci8	Oracle 8
oci8po	Oracle 8 Portable
oracle	Oracle 7 ou menor
postgres7	PostgreSQL 7 ou Acima
pdo_pgsql	PostgreSQL PDO
postgres64	PostgreSQL 6.4 ou acima
postgres	PostgreSQL 6.3 ou abaixo
pdo_sqlite	SQLite PDO
sqlite	SQLite
sybase	Sybase
pdo_sybase_dblib	Sybase PDO DBLIB
pdo_sybase_odbc	Sybase PDO ODBC

Example:

```
$arr_conn = array();
```

```
$arr_conn['drive'] = "mysqlt";
$arr_conn['server'] = "127.0.0.1";
$arr_conn['user'] = "root";
$arr_conn['password'] = "pass123";
$arr_conn['database'] = "sc_samples";
$arr_conn['persistent'] = "Y";
$arr_conn['encoding'] = "utf8";
```

```
sc_connection_new("new_conn_mysql", $arr_conn);
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onCalendarApplicationInit onCalendarScriptInit onScriptInit onLoad onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onScriptInit	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onExecute onLoad	onApplicationInit onFooter onHeader onScriptInit onRecord

sc_date(Date, "Format", "Operator", D, M, Y)

This macro has the aim to calculate the increment or decrement of dates. For its use are necessary:

This macro has the aim to calculate the increment or decrement of dates. For its use are necessary some parameters as follows:

Parameter	Description
Date	Date field that contains the date to be modified.
Format	Formatting that the date field is.
Operator	"+" for increment and "-" for decrement.
D	Number of days to increment or decrement.
M	Number of months to increment or decrement.
Y	Number of Years to increment or decrement.

Ex. 1:

```
{birthdate} = sc_date({birthdate}, "dd/mm/yyyy", "+", 30, 0, 0);
```

Ex. 2:

```
$new_date = sc_date({birthdate}, "yyyy-mm-dd", "-", 15, 3, 2);
```

Ex. 3:

```
{inclusion_dt} = sc_date(date('Ymd'), "yyyymmdd", "-", 0, 1, 1);
```

OBS: To be certified of the date format, before use the function, try it in any event:
echo "form_date = ". {name of the field date};

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onClick onApplicationInit onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onApplicationInit onFooter onHeader onScriptInit onRecord

sc_date_conv({Field_Date}, "Input_Format", "Output_Format")

This macro converts the date field passed as parameter with an input format to another field with an output format.

Parameter	Description
A or Y	Used to reference years.
M	Used to reference Months.
D	Used to reference Days.
Db_Format	Used to get the Database date format.

Ex. 1: Converts the date from "dd/mm/yyyy" format to the "yyyymmdd" format.

```
{field_date} = sc_date_conv({field_date}, "dd/mm/aaaa", "aaaammdd");
```

Ex. 2: Converts the date from the database native format to "dd/mm/yyyy" format.

```
{field_date} = sc_date_conv({field_date}, "db_format", "dd/mm/aaaa");
```

Ex. 3: Converts the date from the "dd/mm/yyyy" format to the Database native format.

```
{field_date} = sc_date_conv({field_date}, "dd/mm/aaaa", "db_format");
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	ReportPDF application
	ajaxFieldonBlur OnChange OnClick				ajaxFieldonBlur OnChange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert		

onExecute	onFocus onAfterDelete onAfterInsert onAfterUpdate onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onFooter onHeader onScriptInit	onClick onApplicationInit onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onFilterInit onFilterRefresh onFilterSave onFilterValidate	onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onFooter onHeader onScriptInit onRecord
-----------	---	--------------------------------------	---	---	--	---	--

sc_date_dif({Date1}, "Format Date1", {Date2}, "Format Date2")

Calculates the difference between two dates (passed as parameters) in days.

Parameter	Description
Date1	Date1 value or variable.
Format Date1	Value or Variable containing the date format stored on Date1.
Date2	Date2 value or variable.
Format Date2	Value or Variable containing the date format stored on Date2.

Ex. 1:
`{amount_days} = sc_date_dif({date1}, "aaaa-mm-dd", {date2}, "mm/dd/aaaa");`

Ex. 2:
`{amount_days} = sc_date_dif("2000-05-01", "aaaa-mm-dd", "04/21/2004", "mm/dd/aaaa");`
`{amount_days}` would be equal to -1451 (days)

Note: *Formats can be different but must be in days, months and years.*

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onClick onApplicationInit onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onApplicationInit onFooter onHeader onScriptInit onRecord

sc_date_dif_2({Date1}, "Format Date1", {Date2}, "Format Date2", Option)

Calculates the difference between two dates (passed as parameters) returning the amount of days, months and years.

The result is returned in an array structure, where the index 0 have the amount of days, the index 1 have the amount of months and the index 2 have the amount of years.

Parameter	Description
Date1	Date1 value or variable.
Format Date1	Value or Variable containing the date format stored on Date1.
Date2	Date2 value or variable.
Format Date2	Value or Variable containing the date format stored on Date2.
Option	1 = Doesn't consider the initial day; 2 = Considers the initial day.

Ex. 1:

```
{differences} = sc_date_dif_2 ({date1}, "yyyy-mm-dd", {date2}, "mm/dd/yyyy", 1);
{dif_days} = {differences[0]};
{dif_months} = {differences[1]};
{dif_years} = {differences[2]};
```

Ex. 2:

```
{differences} = sc_date_dif_2 ("2000-05-01", "yyyy-mm-dd", "04/21/2004", "mm/dd/yyyy", 1);
{differences[0]} = 20 (days)
{differences[1]} = 11 (months)
{differences[2]} = 3 (years)
```

Ex. 3: Using the option 2

```
{differences} = sc_date_dif_2 ("2000-05-01", "yyyy-mm-dd", "04/21/2004", "mm/dd/yyyy", 2);
{differences[0]} = 21 (days)
{differences[1]} = 11 (months)
{differences[2]} = 3 (years).
```

Note: Formats between the two dates can be different but it must have days, months and years

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur OnChange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onClick onApplicationInit onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur OnChange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onApplicationInit onFooter onHeader onScriptInit onRecord

sc_date_empty({Field_Date})

This macro checks if a date field is empty.

Ex. 1:

```
if (sc_date_empty({my_date}))
{
sc_error_message("Invalid Date");
}
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	ReportPDF application
	ajaxFieldonBlur				ajaxFieldonBlur OnChange OnClick OnFocus onAfterDelete		

onExecute	Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onHeader	onClick onApplicationInit onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onFilterRefresh onFilterValidate	onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onLoadAll onRefresh	onHeader onRecord
-----------	--	----------	---	-------------------------------------	---	--	----------------------

sc_decode({My_Field})

Returns, the encrypted field or variable, to its original value.

Ex. 1: Using a local variable:
`{my_field} = sc_decode($field_cript);`

Ex. 2: Using an user variable:
`$my_var = sc_decode($field_cript);`

Note: Netmake isn't responsible by the cryptography algorithm integrity (it is an open source algorithm)

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onClick onApplicationInit onHeader onScriptInit onNavigate onRecord	onApplicationInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onApplicationInit onFooter onHeader onScriptInit onRecord

sc_encode({My_Field})

Returns the field or variable with the content encrypted.

Ex. 1: Encrypting a local variable:
`$field_cript = sc_encode({my_field});`

Ex. 2: Encrypting an user variable:
`$field_cript = sc_encode($my_var);`

Note: Netmake isn't responsible by the encrypting algorithm integrity, (The algorithm provided is open source).

Macro Scope

Blank	calendar	chart	Grid	Search	Form		Menu		ReportPDF
-------	----------	-------	------	--------	------	--	------	--	-----------

application	application	application	application	application	application	Control Form	application	Tree menu	application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onClick onHeader onScriptInit onNavigate onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onApplicationInit onFooter onHeader onScriptInit onRecord

sc_error_continue("Event")

This macro deactivates the Scriptcase standard database error treatment message for the event passed as parameter allowing it to be substituted by the programmer customized error treatment.

It can be used in along with variable: {sc_errro_delete}, {sc_errro_insert} and/or {sc_errro_update}

The events to be passed as parameter are: insert, delete or update.

Ex. 1: Using on the event **onDelete** for the message returned by mssql, trigger treatment, which cancels the record exclusion, displays a error message in the format [Microsoft] [ODBC SQL Server Driver] [SQL Server] You can't erase the record.

```

sc_error_continue("delete");
if (!empty({sc_error_delete}))
{
$tmp = strrpos({sc_error_delete}, "");
if ($tmp !== false)
{
{sc_error_delete} = substr({sc_error_delete}, $tmp 1);
}
}
sc_error_message({sc_error_delete});
sc_error_exit;
}
    
```

Macro Scope

calendar application	Form application	Control Form
onBeforeDelete	onBeforeDelete	onScriptInit
onBeforeInsert	onBeforeInsert	
onBeforeUpdate	onBeforeUpdate	

sc_error_delete

This variable returns a string with a database error message, which occurs by trying to exclude a record.

It is used when there is a need to treat these messages.

Like for example, in case of database validations via triggers or procedures that they return messages.

Ex. 1: Content of the variable {sc_error_delete}, returning an error message generated by database trigger using

```

MsSQL."[Microsoft] [ODBC SQL Server Driver] [SQL Server] You can't erase this register."
    
```

Note: To access the database return error, sees the macro "sc_error_continue".

Macro Scope

calendar application	Form application
onAfterDelete	onAfterDelete

sc_error_exit(URL / My_Application, "Target");

The use of this macro results in the interruption of the application.

This way, any processing that will be executed after the use of the macro or is being executed in the moment will be interrupted.

Here you can see an **example of the macro's using**.

```
This macro should be used after the macro
sc_error_message();
```

This macro can be used in two ways: Without parameters **sc_error_exit()**; that just interrupt the application or informing one URL/application that the user will be redirected when the error is shown.

Macro's parameter, All the parameters are optional.		
Application/URL	Optional	Destiny application's name or URL to where will be redirected.
Target	Optional	_blank, _self

When we inform one destiny application or URL and the Target parameter is not informed the **_self** will be used as default.

Available Targets	
_blank	Show the destiny application or URL in another window or tab.
_self	Show the destiny application or URL in the same window or tab.

Ex. 1: Show only an error message.

```
sc_error_message('Error message!');
sc_error_exit();
```

Ex. 2: Show an error message and an "OK" button to redirect to the URL informed.

```
sc_error_message('Error message');
sc_error_exit(http://www.mysite.com);
```

Ex. 3: Show an error message and an "OK" button to redirect to the application informed.

```
sc_error_message('Error message');
sc_error_exit(my_application, "_blank");
```

Macro Scope

calendar application	Search application	Form application	Control Form
ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onBeforeDelete onBeforeInsert onBeforeUpdate onValidate onValidateFailure onValidateSuccess	onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onNavigate onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess

sc_error_insert

This variable returns a string with the database error, that occurs when adding a record. It is used when is needed to treat these messages. For example, in case of database validations via triggers or procedures that return message.

Ex. 1: Variable **{sc_erro_insert}**, returning an error message generated by database trigger the using mssql

"[Microsoft] [ODBC SQL Server Driver] [SQL Server] You can't add this register."

Note: To access the return of the database error, see the macro **"sc_error_continue"**.

Macro Scope

calendar application	Form application
onAfterInsert	onAfterInsert

sc_error_message("Text")

This macro generates an error messages.

In "**Form and Control**" applications, the messages are presented together with other error messages found in the application, except when using the "**sc_error_exit**" macro.

In "**Grid and Menu**" applications the messages only will be presented through the "**sc_error_exit**" macro.

```

Ex. 1:
if ({discount} > 0.10 && [glo_usr] == 'operator')
{
sc_error_message("Discounting of " . {discount} . " above of the allowed one");
}
    
```

In this example, if the value in the discount field is over 0.10 and the user try to add or save this record, the operation is finished and the error message is presented. The command must finish with ");" (close parentheses and semicolon) used as delimiter for the macro interpreter.

Macro Scope

calendar application	Search application	Form application	Control Form
ajaxFieldonBlur OnChange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onBeforeDelete onBeforeInsert onBeforeUpdate onValidate onValidateFailure onValidateSuccess	onFilterValidate	ajaxFieldonBlur OnChange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onLoad onNavigate onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess

sc_error_update

This variable returns a string with a database error message that occurs when updating a record. It is used when there is a need to treat these messages. For example, in case of database validations via triggers or procedures that return message.

Ex. 1: Variable {**sc_erro_update**}, returning an error message generate by a database trigger using mssql.

```

" [Microsoft][ODBC SQL Server Driver][SQL Server] You can't update this record."
    
```

Note: To have access the return of database error see the macro "**sc_error_continue**".

Macro Scope

calendar application	Form application
onAfterUpdate	onAfterUpdate

sc_exec_sql("SQL Command", "Connection")

This macro allows to condition the circumstances that the SQL commands are executed.

The "Connection" parameter is optional. Required only, if the command is executed in a data base different from the application.

```

Ex. 1:
if (sc_after_delete) {
sc_exec_sql("delete from mytable where key = {key_ant}");
}
    
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
					ajaxFieldonBlur OnChange OnClick				

onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onBeforeDelete onBeforeInsert onBeforeUpdate onCalendarScriptInit onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onFooter onHeader onScriptInit	onClick onGroupBy onHeader onScriptInit onNavigate onRecord	onFilterInit onFilterRefresh onFilterSave onFilterValidate	OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onFooter onHeader onScriptInit onRecord
-----------	---	--------------------------------------	--	---	---	---	--	--	--

sc_exit(Option)

This macro can be used in the ScriptCase events and created button and may return values/process after the execution.

Option	Description
sc_exit()	Returns to the previous application without display anything.
sc_exit(ok)	Displays a window with an OK button adn returns to the previous application.
sc_exit(sel)	Doesn't display the window with an OK button and returns to the current application, repeat the application select.
sc_exit(ref)	Doesn't display the window with an OK button refresh data and returns to the current application.
sc_exit(ok,ref)	Same of the sc_exit(ref) function; however, display the OK button
sc_exit(ok,sel)	Same of the sc_exit(sel) function; however, display the OK button

This option is only valid for Form and Control Applications.

Opton	Description
sc_exit()	Doesn't complete transactions in database.
sc_exit(c)	Commit pending transactions.
sc_exit(r)	Rollback pending transactions.

Macro Scope

calendar application	Form application	Control Form
onScriptInit	onClick onScriptInit	onClick onScriptInit

sc_field_color("Field", "Color")

This macros its used to change/restore a **grid field** text color dynamically.

Ex. 1: Change the grid text field "customer_id" color.

```
sc_field_color ("customerid", "#33FF66");
```

Ex. 2: Change the "value_order" field text color, if the value is greater that 1000.

Otherwise, use the default color.

```
if ({value_order} > 1000)
{
sc_field_color ("value_order", "#33FF66");
}
else
{
sc_field_color ("value_order", "");
}
```

Ex. 3: Using a local variable.

```
sc_field_color ("customerid", {fld_color});
```

Ex. 4: Using a global variable.

```
sc_field_color ("customerid", [glo_color]);
```

Macro Scope

Grid application	ReportPDF application
onScriptInit onRecord	onScriptInit onRecord

sc_field_disabled("Field_Name = True/False", "Parameter")

This macro is used to block a field to get any data that would be typed on it.

Fields that its going to be blocked should be followed by the option "true" (default) or "false".

The parameter its optional, "I" its used only to block de addition of new records, "U" to block only the update, in case of the parameter has not been informed the scriptcase will use both options.

Ex. 1: Blocks only one field for addition and update.

```
sc_field_disabled("Field_01");
```

Ex. 2: Unblocking a field for addition and update.

```
sc_field_disabled("Field_02=false");
```

Ex. 3: Blocks several fields, only for update.

```
sc_field_disabled("Field_01; Field_02; Field_03", "U");
```

Ex. 4: Combination of block and unblock of several fields, for addition and update.

```
sc_field_disabled("Field_01=true; Field_02=false; Field_03=true");
```

Macro Scope

calendar application	Form application	Control Form	ReportPDF application
onLoad	onLoad onNavigate	onLoadAll	onHeader

sc_field_disabled_record("Field_Name = True/False", "Parameter")

This macro has the objective to block the typing on determined fields in the lines of the Multiple Record form, Editable Grid and Editable Grid View.

Fields that are suppose to be blocked should be applied the options, "true" (To block) or "false" (to unblock). If not informed, the default value is "true".

The optional parameter is using the "I" to block while inserting a new record and "U" to block while updating a record. If not informed, Scriptcase will block on both occasions by default.

Ex. 1: Blocking the field while including and updating.

```
sc_field_disabled_record("Field_01");
```

Ex. 2: Unblocking the field while including and updating.

```
sc_field_disabled_record("Field_02=false");
```

Ex. 3: Blocking the various fields only when updating.

```
sc_field_disabled_record("Field_01; Field_02; Field_03", "U");
```

Ex. 4: Combination of blocking and unblocking various fields while including and updating.

```
sc_field_disabled_record("Field_01=true; Field_02=false; Field_03=true");
```

The macro can also work within a condition, blocking a specific field on the record line.

Example 1:

```
if ({{Filed_01}} == 1)
{
sc_field_disabled_record("Filed_03");
}
```

Example 2:

```
switch(trim([arr_apps][ {app_name} ]))
{
case 'form':
break;
case 'calendar':
sc_field_disabled_record("priv_export; priv_print");
break;
case 'cons':
sc_field_disabled_record("priv_insert; priv_delete; priv_update");
break;
default:
sc_field_disabled_record("priv_insert; priv_delete; priv_update; priv_export; priv_print");
break;
}
```

Macro Scope

Form application
onLoadRecord

sc_field_display({My_Field}, on/off)

This macro dynamically display a specific field.
By default all the fields are display ("on" condition).

Ex. 1:

```
if ({tp_customer} == "personal")
{
sc_field_display({company_name}, off);
}
else
{
sc_field_display({personal_name}, off);
}
```

Macro Scope

calendar application	chart application	Grid application	Form application	Control Form	ReportPDF application
ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onHeader onScriptInit	onScriptInit	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeInsertAll onBeforeUpdateAll onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onlick onFocus onScriptInit onLoadAll onRefresh onValidate onValidateFailure	onScriptInit

sc_field_init_off(Field1, Field2,...)

This macro is intended to inhibit the query fields on the initial load

Fields inhibited by this macro can be enabled by using the "Columns" button.

Ex 01:

```
sc_field_init_off ({orderId}, {TotalValue});
```

Macro Scope

Grid application
onApplicationInit

sc_field_readonly({Field}, on/off)

This macro dynamically set a form field attribute to 'ReadOnly'. Use this macro only to do it at 'runtime'. Otherwise set this parameter on / off on the form interface. section "Read-Only".

The on/off parameter its optional, used by an ajax event to define if a field its going to be read-only (On) or read-only (Off).

Ex 1: To set a field as read only even when the form is in "Addition mode":

```
if (sc_btn_new)
{
sc_field_readonly({my_field});
}
```

Ex. 2: To set a field as read only dynamically use:

```
sc_field_readonly({my_field});
```

Macro Scope

calendar application	Form application	Control Form
ajaxFieldonBlur Onchange	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete	

OnClick	onAfterDeleteAll	onBlur
OnFocus	onAfterInsert	onChange
onAfterDelete	onAfterInsertAll	onClick
onAfterInsert	onAfterUpdate	onFocus
onAfterUpdate	onAfterUpdateAll	onScriptInit
onScriptInit	onBeforeInsertAll	onLoadAll
onLoad	onBeforeUpdateAll	onRefresh
onRefresh	onScriptInit	onValidate
onValidate	onLoad	onValidateFailure
onValidateFailure	onNavigate	
onValidateSuccess	onLoadRecord	
	onRefresh	
	onValidate	
	onValidateFailure	
	onValidateSuccess	

sc_field_style({My_Field}, "Background-Color", "Size", "Color", "Family", "Weight")

This macro allows to modify the grid field style dynamically.

Parameter	Description
Field	Name
Background-Color (optional)	Field cell background Color
Size (optional)	Field font size
Color (optional)	Font Color
Family (optional)	Font family
Weight (optional)	Font weight

Ex. 1:

```
sc_field_style({my_field}, '#33FF99', '15px', '#000000', 'Arial, sans-serif', 'bold');
```

Macro Scope

Grid application

onRecord

sc_format_num({My_Field}, "Group_Symb", "Dec_Symb", "Amount_Dec", "Fill_Zeros", "Side_Neg", "Currency_Symb", "Side_Currency_Symb")

This macro is used to format numerical values.

Parameter	Description
My_Field	Variable or field to format (The return is in the next variable).
Group_Symb	Integer values grouping symbol.
Dec_Symb	Decimals separator symbol.
Amount_Dec	Number of decimals to be displayed.
Fill_Zeros	Complete decimals with zeros (fill in) (S = yes and N = no).
Side_Neg	Negative sign position. (1 = left and 2 = right)
Currency_Symb	Monetary symbol to be displayed.
Side_Currency_Symb	Currency symbol position. (1 = left and 2 = right).

Ex. 1: Formatting an integer.

```
sc_format_num({my_value}, ',', '', 0, 'N', '1', '');
Input value = 001234567890
Output value = 1.234.567.890
```

Ex. 2: Formatting a negative value and truncating decimals.

```
sc_format_num({my_value}, ',', '', 0, 'N', '1', '');
Input value = 0012345678.90-
Output value = -12.345.678
```

Ex. 3: Formatting a value with 4 decimals and filling with zeros.

```
sc_format_num({my_value}, ',', '', 4, 'S', '1', '');
Input value = 0012345678.9
Output value = 12.345.678,9000
```

Ex. 4: Formatting a value with 2 decimals, filling with zeros and using monetary symbol.

```
sc_format_num({my_value}, ',', '', 2, 'S', '1', 'US$');
Input value = 0012345678.9
Output value = US$ 12.345.678,90
```

Ex. 5: Formatting a value with 2 decimals, filling with zeros.

```
sc_format_num({my_value}, ',', '', 2, 'S', '1', '');
Input value = .9
Output value = 0,90
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onClick onApplicationInit onFooter onHeader onScriptInit onNavigate onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onExecute onLoad	onApplicationInit onFooter onHeader onScriptInit

sc_format_num_region({My_Field}, "Qtde_Dec", "Insert_Zeros", "Monetary_Sym")

This macro has the objective to format numbers, using the regional settings

Parameter	Description
My_Field	Variable or field that has a value to be formatted (if the value is 0 the result will be the same as the variable).
Qtde_Dec	Amount of decimals that'll be displayed.
Insert_Zeros	Complete with the non existing decimals (Y = yes and N = no).
Monetary_Sym	Display monetary symbol (Y = yes and N = no).

Ex. 1: Formatting a value with 2 decimal spaces, inserting the zeros and the monetary value.

sc_format_num_region({my_value}, "2", "Y", "Y");
 Input value = 9
 Output value = R\$0,90

Macro Scope

Blank application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu
onExecute	onClick onApplicationInit onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad

sc_form_show'on' or 'off'

This macro dynamically display or not a form.

By default, the form is displayed ("on" condition).

Ex. 1: Doesn't display the form.

```
sc_form_show = 'off';
```

Macro Scope

calendar application	Form application	Control Form
onScriptInit onLoad onValidate	onScriptInit onLoad onValidate	onScriptInit onLoadAll

sc_getfield('myField')

This macro is used to assign objects (that represent the field's form) to a javascript variable, allowing the user to access the object properties. This macro will run on control and forms at the javascript events and methods.

EX:

```
Showing the value of the "client" field of a form cli_js = sc_getfield('cliente'); alert (cli_js.value);
```

Macro Scope

sc_get_groupby_rule()

This macro is used to retrieve the name of the Group By rule that is being used at this moment. The return value is a String containing the name of the rule.

Ex:

```
if (sc_get_groupby_rule() == "rule1")
{
[title] = "Sales X Region";
}
```

Macro Scope

chart application	Grid application
onFooter onHeader onScriptInit	onScriptInit onNavigate onRecord

sc_get_language

This macro returns the abbreviation of the language used.

Ex. 1:

```
$my_language = sc_get_language();
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure	onApplicationInit onFooter onHeader onScriptInit	onFooter onScriptInit onNavigate onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onFooter onHeader onRecord

	onValidateSuccess				onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess				
--	-------------------	--	--	--	---	--	--	--	--

sc_get_regional

This macro returns the abbreviation of the regional settings used.

Ex. 1:

```
$my_regional = sc_get_regional();
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onFooter onScriptInit onNavigate onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onHeader onRecord

sc_get_theme

This macro returns the application theme used.

Ex. 1:

```
$my_theme = sc_get_theme();
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onFooter onScriptInit onNavigate onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onApplicationInit onFooter onHeader onScriptInit onRecord

sc_groupby_label("My_Field")

This Macro dynamically modify the field label displayed in group by lines.

Ex. 1:
`sc_groupby_label('state') = "state grouping";`

Ex. 2:
`sc_groupby_label('balance') = "balance total";`

Macro Scope**Grid application**

onGroupBy

sc_hide_groupby_rule('group1', 'grop2', 'group3')

This macro is used to disable Group By rules in execution time.

The Group By rules ID should be informed as parameters in the macro.

Ex:

```
if( [usr_login_group] == 'seller' ){
sc_hide_groupby_rule('rule_1', 'rule_2');
}
```

Macro Scope**chart application** **Grid application**

onApplicationInit	onApplicationInit
-------------------	-------------------

sc_image(Image01.jpg)

This macro is used to copy images in some events or Javascript commands to load images to an application.

All images used in the applications are automatically copied to each application.

Ex. 1:
`sc_image (img1.gif, img2.jpg);`

Note: *The images must exist in the development environment images directory (../devel/conf/sys/img/img).*

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	onApplicationInit onScriptInit onLoad onValidate	onApplicationInit onScriptInit	onFooter onScriptInit	onFilterInit	onApplicationInit onClick onScriptInit onLoad onLoadRecord onValidate	onApplicationInit onScriptInit onLoadAll onValidateFailure onValidateSuccess	onApplicationInit onLoad	onLoad	onApplicationInit onScriptInit

sc_include("File", "Source")

The Scriptcase allows the user to catalogue his PHP scripts to use in various applications.

To catalogue the scripts go to the ScriptCase main menu at **"Tools > Library"** and use the "upload" option existing in the interface.

When a script is catalogued, the user has the option of the context choice where it will be stored, global level (Scriptcase), group level or user level.

- **Public:** All the projects will have access.
- **Project:** Only the project users (the one's that will be logged at that time) will have access.
- **User:** Only the user (the one's that will be logged at that time) will have access.

The **sc_include** macro, the user must inform the **script name** that will be included and the **source**.

- **Script Name:** Script name to be included.

The Source context that the script was saved, that could be:

- **Sys or Blank:** Scriptcase global level.
- **Prj:** Group level.
- **Usr:** User level.

Ex. 1: Including a catalogued script to global level.

```
sc_include(my_script.php);
```

Ex. 2: Including a catalogued script to the group level.

```
sc_include(my_script.php, grp);
```

Ex. 3: Including a catalogued script to the user level.

```
sc_include(my_script.php, usr);
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onHeader onScriptInit	onScriptInit onNavigate onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onExecute onLoad	onApplicationInit onHeader onScriptInit onRecord

sc_include_lib("Lib1", "Lib2", ...)

This macro its used to select dynamically the application libraries.

PHP Libraries	JavaScript Libraries
tcpdf	jquery
fpdf_html	jquery_thickbox
excel	jquery_blockui
rtf	tiny_mce
-	jquery_touch_punch
-	fusioncharts
-	jsmind
-	jkanban
-	orgchart

Ex. 1:

```
sc_include_lib ("tcpdf");

$pdf=new TCPDF();
$pdf->AddPage();
$pdf->SetFont('Arial','B',16);
$pdf->Cell(40,10,'Hello World!');
$pdf->Output();
```

Ex. 2:

```
?>
sc_include_lib ("jsmind");
< ?php
```

Macro Scope

Blank application	Grid application
onExecute	onScriptInit

sc_include_library("Target", "Library Name", "File", "include_once", "Require")

This macro includes a **PHP** file from a library in the application. You must have a library created in Scriptcase to make use of this macro. To create a library, go to "Tools -> Libraries".

Parameter	Description
Target	Tells you what the <i>scope</i> of the library. Can be "sys" for libraries of <i>Public</i> scope or "prj" for libraries of <i>Project</i> scope.
Library Name	Name given to the library at the time of creation.
File	Absolute path within the library.
include_once (optional)	Make sure that the file will only be included once. If not informed, this value is set as "true".
Require (optional)	If the file or library does not exist, the application to be executed and returns an error. If not informed, the value is set as "true".

Ex: Including a file from a library:

```
sc_include_library("prj", "phpqrcode", "qrlib.php", true, true);
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu
	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onScriptInit onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onExecute onLoad

sc_label({My_Field})

This macro its used to modify dynamically the grid form field label.

Ex. 1: Using the field name.

```
sc_label("Customer") = "Cust Name";
```

Ex. 2: Using a variable as paramenter.

```
sc_label({Customer}) = "Cust Name";
```

Ex. 3: Using a global variable.

```
sc_label([global_variable]) = "Cust Name";
```

Macro Scope

calendar application	chart application	Grid application	Form application	Control Form	ReportPDF application
onScriptInit onLoad onRefresh	onScriptInit	onClick onScriptInit onNavigate	ajaxFieldonBlur Onchange OnClick OnFocus onScriptInit onLoad onNavigate onLoadRecord	onBlur onChange onClick onFocus onScriptInit onLoadAll	onScriptInit

sc_language

This macro returns the language and regional settings.

Ex. 1:

```
echo "Language: ".{sc_language};
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	onApplicationInit onCalendarApplicationInit onCalendarScriptInit onScriptInit	onApplicationInit onScriptInit	onScriptInit	onApplicationInit onFilterInit	onApplicationInit onScriptInit	onApplicationInit onScriptInit onValidate onValidateSuccess	onApplicationInit onLoad	onApplicationInit onLoad	onApplicationInit onScriptInit

sc_ldap_login

\$ldapConn = sc_ldap_login(\$server [, \$domain, \$user, \$password, \$dn, \$group]);

This macro initiates the connection and bind it to the user credentials. This macro needs the following parameters:

Server =
Domain =
User =
Password =
DN =
Group =

There are three possible returns:

Macro return	Situation
(string)'ldap_connect_error'	Connection error with LDAP host/service.
(string)'ldap_bind_error'	Authentication error after successful connection.
(resource)	Successful connection and bind.

Macro Scope

calendar application	Form application	Control Form
onApplicationInit onValidate	onApplicationInit onValidate	onApplicationInit onValidate

sc_ldap_logout()

This macro is used to release the connection after using the macro sc_ldap_login.

Ex1:

```
sc_ldap_logout();
```

Macro Scope

calendar application	Form application	Control Form
onApplicationInit onValidate	onApplicationInit onValidate	onApplicationInit onValidate

sc_ldap_search(\$filter = 'all', \$attributes = array())

Macro to perform searches in the LDAP.

Parameter	Description
\$filter	'all' or 'member': 'all' returns all information that the logged user can see. 'member' returns the information about that member
\$attributes	Array of LDAP filter options.

Ex:

```
$result = sc_ldap_search($filter, array('mail', 'displayname'));
```

Macro Scope

calendar application	Form application	Control Form
onApplicationInit	onApplicationInit	onApplicationInit
onValidate	onValidate	onValidate

sc_link(Column, Application, Parameters, "Hint", "Target", Height, Width)

This macro dynamically creates or modifies links between grid applications and other applications.

Column: Grid column field name to generate the link.

Application: Application name to establish the link.

Parameters: Parameters required by the called application.

Must be informed "=" (attribution sign) . More than one parameter must be separated by ";" (semicolon).

Hint: Observation or tip displayed when the mouse cursor passes over the link.

Target : Where the application will function call and may be:

Target	Descrição
"_self"	For carrying on the same page (overrides the query);
"_blank"	To run on another page (pop-up);
"iframeT"	To run in an iframe above the Grid (top);
"iframeL"	To run in an iframe, the left of the Grid (Left);
"iframeR"	To run in an iframe, the left of the Grid (Left);
"iframeB"	To run in an iframe below the Grid (Bot);
"modal"	Opens a new window and blocks any interaction in the main window;

In the case of modal target, there are two optional parameters *:

***Height:** Set the height of the modal window (default 440);

***Width:** Set the width of the modal window (default 630);

Note: If omitted, will be assigned the value "_self".

Ex. 1: Creates a link to the customer.php application, on the Id field, passing the global variable [global_costumer] as and the product field (local variable) {Id} to the customer.php application:

```
sc_link(Id, customer.php, product=[global_costumer]; Id={Id}, "Client Data", "_blank");
```

Ex. 2: Creating a dynamic link to another application depending on the field valor {customer_type} and opening it in a modal.

```
if ({customer_type} == "F") {
sc_link (customer_type, type_f.php, product=[global_costumer]; Id={Id}, " Personal Data ", "modal",
500, 700);
}
else{
sc_link (customer_type, type_a.php, product=[global_costumer]; Id={Id}, " Company Data ",
"iframeR");
}
```

Ex. 3: Creating a link to an URL: in this case, "paramater and target" have no effect. It will be executed redirecting to a informed URL.

```
sc_link (my_field, http://www.scriptcase.com, ,"hint of the link");
```

Macro Scope

Grid application	ReportPDF application
onRecord	onRecord

sc_log_add("action", "description")

This macro is intended to add a record to the log table, thereby making the message and / or the name of the action customizable to the developer. The macro parameters can use PHP variables.

Ex:

```
sc_log_add ("access", "Application has been accessed by the user trial");
```

Macro Scope

onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onBeforeDelete onBeforeInsert onBeforeUpdate onCalendarScriptInit onScriptInit onLoad onRefresh onValidate onValidateSuccess	onFooter onHeader onScriptInit	onClick onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateSuccess	onBlur onChange onClick onFocus onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onFooter onHeader onScriptInit onRecord
-----------	--	--------------------------------------	--	---	--	--	--	--	--

sc_mail_send(SMTP, Usr, Pw, From, To, Subject, Message, Mens_Type, Copies, Copies_Type, Port, Connection_Type, Attachment, SSL)

This macro is used to send emails.

Parameter	Description
SMTP	SMTP server name or IP address. (String or Variable that contains the server name)
Usr	SMTP user name. (String or Variable that contains the user name)
Pw	SMTP password. (String or Variable that contains the password)
From	From email. (String or Variable that contains the email)
To	List of the emails that will receive the message, it could be a string or variable that contains one or more emails separated by ";" or one variable that contains one array of emails.
Subject	Message subject. (String or Variable that contains the subject)
Message	Message body. (String or Variable that contains the message)
Mens_Type	Message format: (T)ext or (H)tml.
Copies	List of the emails that will receive the message, it could be a string or variable that contains one or more emails separated by ";" or one variable that contains one array of emails.
Copies_Type	Type copies: BCC (Hidden copies) or CCC (Regular copies).
Port	The port used by your e-mail server. Use the port 465 for the security type SSL, and the port 587 for the security type TLS or the port 25 for no security. If not informed, Scriptcase will set the default port: 25
Connection_Type	Indicating a secure connection. Use S for SSL, T for TLS or N for non secure connections. If not informed, Scriptcase will set the default value: N .
Attachment	Absolute path of the file that will be attached in the email
SSL	Optional parameter to add SSL settings. If you need to add more than one setting, use "," for separate them. Ex: \$ssl = 'allow_self_signed=false;verify_peer=false';

Note: To use safe connection **its required** to have the PHP "**php_openssl**" extension enable.

Ex. 1:

```
sc_mail_send('smtp.meuserver.com.br', 'usr', 'pw', 'from@netmake.com.br', 'to@netmake.com.br',
'Test Email Subject', 'Message', 'H', 'abc@cop.com.br;zyx@cop.com.br', 'H', "", "", 'c:/test/arq.txt');
```

Ex. 2: Using variables and defining the port.

```
sc_mail_send([glo_smtp], [usuar], [glo_pw], {from}, {to}, {subject}, {message}, 'T', [glo_copias],
'CCC', '419', 'S', {attached});
```

Ex. 3: No user and password.

```
sc_mail_send([glo_smtp], "", "", 'from.netmake.com.br', {to}, {subject}, {message}, 'H', "", "", "",
[glo_att]);
```

The amount of emails sent will be stored at the special variable "sc_mail_count".

If any error happen when the scriptcase try to send the email, the variable "sc_mail_ok" will receive "false" and the error message will be stored on the variable "sc_mail_erro".

Ex.4 :

```
sc_mail_send([glo_smtp], "", "", 'from@netmake.com.br', {to}, {subject}, {message}, 'H');
if ({sc_mail_ok})
```

```

{
echo "sent {sc_mail_count} e-mail(s) with success!!";
}
else
{
sc_error_message({sc_mail_erro});
}
}

```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	onAfterDelete onAfterInsert onAfterUpdate onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onFooter onHeader onScriptInit	onClick onHeader onScriptInit onRecord	onFilterInit onFilterSave onFilterValidate	onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onFooter onHeader onScriptInit onRecord

sc_make_link(Application, Parameters)

This macro creates a string containing a link to other Scriptcase application (when writing html code).

Parameter	Description
Application	Name of the application to establish the link.
Parameters	Parameters necessary so called application. In the format: "=" (the attribution sign) . More than one parameter must be separated by ";" (semicolon).

Ex. 1: Creating a string with a link for the datacli.php application, without parameters passage.

```
$string_link = sc_make_link(datacli.php);
```

Ex. 2: Creating a string with a link for the application datacli.php, with parameters passage.

```
$string_link = sc_make_link(datacli, parm1={company};parm2=[glo_office]);
```

Macro Scope

Blank application	chart application	Grid application	Control Form
onExecute	onFooter onHeader onScriptInit	onScriptInit onNavigate onRecord	onValidate onValidateSuccess

sc_master_value('Object', Value)

Update a data in the **master** application, in real time, according to changes made in a given field of the **detail** application.

Parameter	Description
Object	Object name to be updated in master Application. It's not necessary the {} in a field name.
Value	Object value. The value can be a field or a variable.

Ex. 1:

```
sc_lookup(result,"SELECT SUM(Total) FROM adm_order_itens WHERE OrderID = '{OrderID}'");
```

```
//Using sc_format_num() to format the value that will be updated in the master application
$value = sc_format_num({result[0][0]}, ',', '.', 2, 'S', 1, '');
sc_master_value('OrderPrice', $value);
```

* If the value, that will be sent to the master, be numeric, you have to use **sc_format_num()** to format it. To see the macro documentation, [click here](#).

Macro Scope

calendar application	Form application	Control Form
onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure	onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure	onScriptInit onLoadAll onRefresh

sc_menu_delete(Id_Item1)

This macro removes menu items passed as parameters. Can be use in menu applications or tree menu.

Observe that the parameters used in this macro are the "ID" of the items.

If a menu item its removed and it have "nodes", they will be hidden too.

Macro Scope

Menu application	Tree menu
onApplicationInit onLoad	onApplicationInit onLoad

sc_menu_disable(Id_Item1)

This macro deactivates menu items pass as parameters. Can be used in menu applications or tree menus.

The parameters passed are items "ID". Menu items are deactivated recursively (when a submenu item is deactivated its "nodes" are also deactivated).

Macro Scope

Menu application	Tree menu
onApplicationInit onLoad	onApplicationInit onLoad

sc_menu_force_mobile(boolean)

This Macro is used to force the creation of menus to mobile devices.

The parameter (true or false) is optional. If no value is passed, it will use the value "true" (enable mobile mode)

Ex1:

```
sc_menu_force_mobile(true);
```

Ex2:

```
sc_menu_force_mobile(false);
```

Macro Scope

Menu application	Tree menu
onLoad	onApplicationInit onLoad

sc_menu_item

This variable is onlv available for menu application and it has the obiective of aiving access to a

menu item that was selected, this way the user can take decisions before to run the application.

Ex. 1: To clean global variable.

```
if ({sc_menu_item} == "item_1")
{
sc_reset_global([global1], [global2]);
}
```

Ex. 2: If an application requires a global variable another application can initialize it accordingly.

```
if ({sc_menu_item} == "item_5" && (!isset([glo_employ]) || empty([glo_employ])))
{
sc_redir(apl_inf_emp);
}
```

Ex. 2: Use the button on the toolbar menu.

```
if ({sc_menu_item} == "btn_1")
{
sc_apl_conf("form_customer", "start", "new");
}
```

Macro Scope

Menu application	Tree menu
onExecute	onApplicationInit onExecute

sc_redir(Application, Parameter01; Parameter02; Target, Error, height_modal, width_modal)

This macro its used to redirect the processing to other application or URL.

If the redir uses parameters, these must be passed in the following format:

- 1) After the name of the application, use the comma delimiter (,) then
- 2) = (equal sign) .
- 3) More than one parameter, must be separated by semicolon (;)
- 4) The target determine which application is opened (default=_self): _self, _parent, _blank or modal.
- 5) Optional parameter to set error messages redirection in the application: "F" redirects if there is error in the application (default value) and "E" does not redirect.

OBS. If your connection uses transaction control the use of this macro on the events onAfterInsert, onAfterUpdate, onAfterDelete, onBeforeInsert, onBeforeUpdate or onBeforeDelete should come after the use of macro sc_commit_trans so this way it will save the form changes

This macro does not support the use of the Modal parameter when used in a Run button.

Ex. 1: Application without parameters nor target.

```
if ([global_user] == "test")
{
sc_redir(application_x.php);
}
```

Ex. 2: Application with parameters and without target.

```
if ([global_user] == "test")
{
sc_redir(application_x, parm1={var_test}; parm2="xxx");
}
```

Ex. 3: Application without parameters and with target.

```
if ([global_user] == "test")
{
sc_redir(application_x, "", "_parent");
}
```

Ex. 4: Application with parameter and target.

```
if ([global_user] == "test")
{
sc_redir(application_x, parm1={var_test}; parm2="xxx", "_blank");
}
```

Ex. 5: URL.

```
if ([global_user] == "test")
{
sc_redir(http://www.my_page.com);
}
```

OBS. Everything that was passed as parameter to the called application will be available as global variable.

EX: sc_redir(employee.php, parm1={var_test}; parm2="xxx", "_blank");

onExecute	onApplicationInit onCalendarApplicationInit onCalendarScriptInit onScriptInit	onApplicationInit onScriptInit	onApplicationInit onScriptInit	onApplicationInit onFilterInit	onApplicationInit onScriptInit	onApplicationInit onScriptInit onValidate onValidateSuccess	onApplicationInit onLoad	onApplicationInit onLoad	onApplicationInit onScriptInit
-----------	--	-----------------------------------	-----------------------------------	-----------------------------------	-----------------------------------	--	-----------------------------	-----------------------------	-----------------------------------

sc_reset_change_connection

This macro has the objective to erase the changes made using the macro "sc_change_connection".

Ex. 1:

```
sc_reset_change_connection();
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	onApplicationInit onCalendarApplicationInit onCalendarScriptInit onScriptInit	onApplicationInit onScriptInit	onApplicationInit onScriptInit	onApplicationInit onScriptInit	onApplicationInit onScriptInit onValidate onValidateSuccess	onApplicationInit onLoad	onLoad	onScriptInit

sc_reset_connection_edit

This macro undoes the connection edits made by macro "sc_connection_edit".

Ex. 1:

```
sc_reset_connection_edit();
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur OnChange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onCalendarApplicationInit onCalendarScriptInit onScriptInit onLoad onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onScriptInit	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur OnChange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onExecute onLoad	onApplicationInit onFooter onHeader onScriptInit onRecord

sc_reset_connection_new

This macro undoes the connections made by the macro "sc_connection_new".

```
sc_reset_connection_new();
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur OnChange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert	onApplicationInit onFooter onHeader onScriptInit	onScriptInit	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur OnChange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll	onApplicationInit onExecute onLoad	onExecute onLoad	onApplicationInit onFooter onHeader onScriptInit onRecord

onBeforeUpdate onCalendarApplicationInit onCalendarScriptInit onScriptInit onLoad onValidateSuccess					onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onValidateSuccess	onRefresh onValidate onValidateFailure onValidateSuccess			
--	--	--	--	--	--	---	--	--	--

sc_reset_global([Global_Variable1], [Global_Variable2] ...)

This macro has the objective to delete global variables, stored in the PHP session.

Ex. 1:
sc_reset_global ([Login], [Pass]);

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onCalendarApplicationInit onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onClick onApplicationInit onGroupBy onHeader onScriptInit onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onLoad	onApplicationInit onFooter onHeader onRecord

sc_reset_menu_delete

This macro its used to restore a menu item, deleted by the macro "sc_menu_delete".

Ex. 1:
sc_reset_menu_delete();

Macro Scope

Menu application	Tree menu
onApplicationInit onLoad	onApplicationInit onLoad

sc_reset_menu_disable

This macro have the objective to enable a menu item, disabled by the macro "sc_menu_disable".

Ex. 1:
sc_reset_menu_disable();

Macro Scope

Menu application	Tree menu
onApplicationInit onLoad	onApplicationInit onLoad

sc_rollback_trans("Connection")

This macro its used to cancel a transaction set in the database.

The "Connection" parameter is optional, use only if the command is executed in a different data base from the specified to the application.

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onHeader onScriptInit	onClick onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onClick onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onExecute onLoad	onHeader onRecord

sc_script_name

This macro is available to the MENU applications to access the application name that was selected in the menu. It allows the user to make decisions before executing the application if a pre-process is required to that application.

Ex. 1: To clean global variables.

```
if ({sc_script_name} == "ap1")
{
sc_reset_global([global1], [global2]);
}
```

Macro Scope

Menu application	Tree menu
onExecute	onExecute

sc_select(dataset, "SQL Command", "Connection")

This macro executes the SQL commands passed as parameter and access the "dataset" in the command.

Different from sc_lookup macro, this macro doesn't manipulate the dataset (the user is responsible for all the manipulation).

If an error occurs in the sql command execution, the variable attributed to the database returns as "false" and the error message is available in the "dataset_error" variable.

The connection parameter is optional, use only if the command is executed in a data base different from the specified in the application.

Ex. 1:

```
sc_select(my_data, "select clientId, clientName, limitecred from costumers");
if ({my_data} === false)
{
echo "Access error. Message =". {my_data_errro};
}
else
{
while (!{my_data}->EOF)
{
{clientName} = {my_data}->fields[1];
{my_data}->MoveNext();
}
{my_data}->Close();
}
```

Ex. 2: The SQL command can be passed as application fields (local variables) or of global variables.

```
sc_select(dataset,"select price order from order where clientId = '{clientId}' and cod_Seller = [var_glo_seller]");
```

Note: The command must always be finished with semicolon";".

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onBeforeDelete onBeforeInsert onBeforeUpdate onCalendarScriptInit onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onFooter onHeader onScriptInit	onClick onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onExecute onLoad	onFooter onHeader onRecord

sc_select_field({Field})

This macro modify dynamically the SELECT: Is possible to modify dynamically the grid fields through the dynamic modification of the original select.

Ex. 1: A table with the columns year and the months from January to December is used to create a grid using a parameter, displays the years and only a specific month. Passing the parameter as a global variable named "show_month", in the application is found the following structure:

Select Clause: select year, pair_month from table

```
sc_select_field(pair_month) = [show_month];
sc_label(pair_month) = [show_month];
Modify the select, and the grid column label.
```

It is possible, during the grid execution time, modify the grid ORDER BY clause. At least one ORDER BY is required in the original grid SQL command to allow the substitution execution during the grid time.

Ex. 2: Select Clause: select code, name, value from table order by code. To change the order, from code to name, use: sc_select_order("code") = "name";

Note: This command must be in the context of the event "onInit";

Macro Scope

chart application	Grid application
onScriptInit	onScriptInit

sc_select_order("Field")

It is possible, during the grid execution time, modify the grid ORDER BY clause. At least one ORDER BY is required in the original grid SQL command to allow the substitution during the grid execution time.

Ex. 1: Select Clause: SELECT code, name, price FROM table ORDER BY code To change the order from code to name, we will have: sc_select_order("code") = "name";

Ex. 2: Select Clause: SELECT code, name, price, date FROM table ORDER BY code, name, price To change the order from name to date, we will have: sc_select_order("name") = "date";

Note: This command must be in the context of "process before the select".

Macro Scope

chart application	Grid application
onScriptInit	onHeader onScriptInit

sc_select_where(add)

Is possible, during grid execution time to add a field/condition to the search WHERE clause.

Ex. 1: It adds the contents between the quotes (") in the grids select.

```
if (empty({sc_where_current}))
{
sc_select_where(add) = "where campoX > [global]";
}
else
{
sc_select_where(add) = "AND campoX > [global_variable]";
}
```

Note: This command must be in the context of "process before the select".

Macro Scope

chart application	Grid application
onScriptInit	onHeader onScriptInit

sc_send_mail_api(\$arr_settings)

This macro allows dynamic sending of embedded emails with the **Mandrill** and **Amazon SES** APIs. It is also possible to send e-mail using the **SMTP** protocol.

This macro uses the same mandrill parameters to send e-mail, regardless of the Gateway (Changer, Amazon SES or SMTP) selected. [click here](#) and see the list of parameters that can be used.

This macro contains a single parameter "**\$arr_settings**", which receives an array with the email sending information.

For more information on the parameters the mandrill [click here](#)

Parameter	Description
\$arr_settings	Array with information for use of APIs.

Array indices	Description
profile	Name of some API already configured in "Tools> API", as can be seen in Example 1. When informing a profile, the settings index should not be informed.
settings	E-mail sending server settings. The information must conform to the API used. Note: When informing the settings, the Profile index is disregarded.
message	Message that will be sent by email. Subject, body, to, etc.

Example 1: Sending e-mail using a pre-configured Profile in "Tools> API".

```
if({api_type} == 'smtp' ){
$var_config = array(
'profile' => 'yahoo_example',

'message' => [
'html' => {email_body},
'text' => "",
'to' => $arr_merge,
'subject' => {subject}
]
);
}
sc_send_mail_api($var_config);
```

Example 2: Sending e-mail via SMTP

```
$smtp_server = "smtp.mail.yahoo.com";
$smtp_port = "465";
$smtp_user = "scriptcase.export";
$smtp_password = "scriptcase.export";
$from_email = "scriptcase.export@yahoo.com";
$from_name = "Scriptcase Export";

if ( {api_type} == 'smtp' ){
$var_config = array(

'profile' => "",

'settings' => [
'gateway' => 'smtp',
'smtp_server' => $smtp_server,
```

```
'smtp_port' => $smtp_port,
'smtp_user' => $smtp_user,
'smtp_password' => $smtp_password,
'from_email' => $from_email,
'from_name' => $from_name
},
'message' => [
  'html' => {email_body},
  'text' => '',
  'to' => $arr_merge,
  'subject' => {subject}
]
);
}
```

```
sc_send_mail_api($var_config);
```

Example 3: Sending email using Mandrill API

```
if({api_type} == 'mandrill'){
$var_config = array(
'settings' => [
'gateway' => 'mandrill',
'api_key' => {api_key},
'from_email' => {from_email},
'from_name' => {from_name}
],
'message' => [
'html' => {email_body},
'text' => '',
'to' => array($arr_merge),
'subject' => {subject}
]
);
}
```

```
sc_send_mail_api($var_config);
```

Example 4: Sending email using Amazon SES API

```
if({api_type} == 'ses'){
$var_config = array(
'settings' => [
'gateway' => 'ses',
'region' => {ses_region},
'api_key' => {api_key},
'api_secret' => {ses_secret},
'from_email' => {from_email}
],
'message' => [
'html' => {email_body},
'text' => '',
'to' => array($arr_merge),
'subject' => {subject}
]
);
}
```

```
sc_send_mail_api($var_config);
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	onAfterDelete onAfterInsert onAfterUpdate onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onFooter onHeader onScriptInit	onClick onHeader onScriptInit onRecord	onFilterInit onFilterSave onFilterValidate	onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onFooter onHeader onScriptInit onRecord

sc_send_sms(\$arr_settings)

This macro receives an array as the configuration parameter that contains the message, recipient, gateway, and API key.

Parameter	Description
\$arr_settings	Array with the information for use in the macro.

Array Index	Description
to:	SMS recipient
message:	The content of the text message
gateway:	API to be used (clickatell, plivo or twilio)

auth_token: API Authentication Token

Sample 1 - Basic sending Structure:

```
$arr_settings = array(
    'message' =>
    [
        'to' => '+559999999999',
        'message' => 'Scriptcase SMS'
    ],
    'settings' =>
    [
        'gateway' => 'clickatell',
        'auth_token' => '54Cik1-gQ4GqtuN70d4y4w=='
    ]
);
```

sc_send_sms(\$arr_settings);

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onCalendarApplicationInit onCalendarScriptInit onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onClick onApplicationInit onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onApplicationInit onFooter onHeader onScriptInit onRecord

sc_send_whatsapp(\$arr_settings)

This macro receives an array as a setting parameter that contains the API name setted in the Scriptcase, the message and the destiny number.

The parameters that will be used are:

- **profile:** Here will be informed the API name configured into the Scriptcase interface.
- **message:** In this parameter is informed the message that will be sent through the macro.
- **to:** Allows informing the destiny number of the message.

Exemplo de uso:

```
$arr_settings = array(
    'profile' => 'ChatAPI',
    'message' => 'Test whatsapp',
    'to' => '558182499699'
);

sc_send_whatsapp($arr_settings);
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onCalendarApplicationInit onCalendarScriptInit onScriptInit onLoad onRefresh onValidate onValidateFailure	onApplicationInit onFooter onHeader onScriptInit	onClick onApplicationInit onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onApplicationInit onFooter onHeader onScriptInit onRecord

	onValidateSuccess				onRefresh onValidate onValidateFailure onValidateSuccess				
--	-------------------	--	--	--	---	--	--	--	--

sc_seq_register

This macro provide the record (being processed) grid sequential number. It can be used, only in **onRecord** event.

```

Ex. 1:
if({sc_seq_register} == 10) {
// last line of the page
{total} = {sum_total};
}
    
```

Macro Scope

Grid application
onRecord

sc_set_fetchmode(parm);

This macro allows to change the type of return from the dataset of the select commands. The macro will processed before running the SQL command. Informing the parameter, the dataset will return the index or the name of the column.

- parm = 0**: Returns an array with the index and the name of columns (Scriptcase's Default)
- parm = 1**: Returns an array only with indexes of the columns

Example 1: Returning an array with the name of the columns.

```

sc_set_fetchmode(0);
sc_select(my_data, "SELECT customerid, stateid, birthdate, creditlimit FROM customers where customerid = 'ALFKI'");

while (!$my_data->EOF) {
$meus_dados->fields['customerid'];
$customer = $my_data->fields['stateid'];
$customer = $my_data->fields['birthdate'];

$my_data->MoveNext();
}
$my_data->Close();
    
```

Example 2: Returning an array with the index of the columns.

```

sc_set_fetchmode(1);
sc_select(my_data, "SELECT customerid, stateid, birthdate, creditlimit FROM customers where customerid = 'ALFKI'");

while (!$my_data->EOF){
$my_data->fields[1];
$customer = $my_data->fields[2];
$customer = $my_data->fields[3];

$my_data->MoveNext();
}
$my_data->Close();
    
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure	onFooter onHeader onScriptInit	onClick onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad	onBlur onChange onClick onFocus onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onExecute onLoad	onExecute onLoad	onFooter onHeader onScriptInit onRecord

	onValidateSuccess				onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess				
--	-------------------	--	--	--	---	--	--	--	--

sc_set_focus('Field')

This macro its used to set the focus to a form field.

Ex. 1:

```
sc_set_focus('name');
```

Macro Scope

calendar application	Form application	Control Form
OnClick	OnClick	onApplicationInit
onLoad	onLoad	onScriptInit
onRefresh	onRefresh	onLoadAll
onValidateSuccess	onValidateSuccess	onRefresh

sc_set_global(\$variable_01) or ({My_Field})

This macro its used to register global variables. Create a session variable with the same name and content of the local variable.

NOTE: This macro will be discontinued soon. You should use the method of creating global variables using brackets. Ex: [var_glo_user] = "test";

Ex. 1: Registering a users variable.

```
$my_var = "mary";  
sc_set_global($my_var);
```

Creates a session variable with the name "my_var" with the "mary" content.

Ex. 2: For register a field value as a session variable.

```
sc_set_global({fieldname});
```

Note: This macro doesn't attribute values. Only register the session variables from PHP.

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur OnChange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onCalendarApplicationInit onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onClick onApplicationInit onFooter onGroupBy onHeader onScriptInit onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur OnChange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onFooter onHeader onScriptInit onRecord

sc_set_groupby_rule

This macro is used to select a GROUPBY rule in execution time in the onApplicationInit event.

Ex:

```
sc_set_groupby_rule = "rule id";
```


Macro Scope

chart application	Grid application
onApplicationInit	onApplicationInit

sc_set_language('String Language')

This macro allows to define the application language dynamically.

Ex. 1: Changes the language to English.

```
sc_set_language('en_us');
```

Ex. 2: Changes the language to Spanish.

```
sc_set_language('es');
```

It still allows to define the language and regional settings of the applications using a delimiter ";" between the parameters.

Ex. 1: Changing the language to English and regional settings to Canada.

```
sc_set_language('en_us;en_ca');
```

Ex. 2: Changing the language to Spanish and regional settings to Mexico.

```
sc_set_language('es;es_mx');
```

Ex. 3: Changing the language to Germanian and regional settings to Luxembourg.

```
sc_set_language('de;de_lu');
```

Ex. 4: Changing the language to Slovak and regional settings to Slovak Republic.

```
sc_set_language('sk;sk_sk');
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	onApplicationInit onScriptInit	onApplicationInit onScriptInit	onApplicationInit onScriptInit	onApplicationInit onFilterInit onFilterValidate	onApplicationInit onScriptInit	onApplicationInit onScriptInit onValidate onValidateSuccess	onApplicationInit onLoad	onApplicationInit onLoad	onApplicationInit onScriptInit

sc_set_pdf_name"sc_my_file_name.pdf"

The resulting files of the exported data (PDF, XLS, XML, CSV, WORD and RTF) are stored in "temporary" directory specified in the Scriptcase's configuration in development and in production, with the prefix "sc_" followed by a unique identification so the file of a user does not overwrite to one another.

The prefix "sc_" makes Scriptcase delete the old files. To assign different names Scriptcase patterns, can use the following macros:

- sc_set_csv_name
- sc_set_pdf_name
- sc_set_rtf_name
- sc_set_xml_name
- sc_set_xls_name
- sc_set_word_name

Ex1:

```
sc_set_xls_name = "my_file.xls";
```

```
Ex2: sc_set_csv_name = "sc_my_new_file.csv";
```

NOTE:

- 1) These macros should be used in the event "onApplicationInit."
- 2) Even using the macro, remained the prefix "sc_" (as Ex2), the Scriptcase will delete the file when it reaches the configured timeout limit (see the setting "File Lifetime" in the production environment).

Macro Scope

chart application	Grid application
onApplicationInit	onApplicationInit

sc_set_regional('String Regional')

This macros allows to dynamically change the application regional settings.

Ex. 1:

```
sc_set_regional('en_us');
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	onApplicationInit onScriptInit	onApplicationInit onScriptInit	onApplicationInit onScriptInit	onApplicationInit onFilterInit onFilterValidate	onApplicationInit onScriptInit	onApplicationInit onScriptInit onValidate onValidateSuccess	onApplicationInit onLoad	onApplicationInit onLoad	onApplicationInit onScriptInit

sc_set_theme('String Theme')

This macros allows to define the application theme dynamically.

Ex. 1:

```
sc_set_theme('Newyellow');
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	onApplicationInit onScriptInit	onApplicationInit onScriptInit	onApplicationInit onScriptInit	onApplicationInit onFilterInit onFilterValidate	onApplicationInit onScriptInit	onApplicationInit onScriptInit onValidate onValidateSuccess	onApplicationInit onLoad	onApplicationInit onLoad	onApplicationInit onScriptInit

sc_site_ssl

This macro is used to verify if a used site is a safe site. (https protocol)

Ex. 1: Using an affirmative.

```
if (sc_site_ssl)
{
echo "ok - Safe site";
}
```

Ex. 2: Using a negative.

```
if (!sc_site_ssl)
{
echo "Warning - Unsafe site";
}
```

Ex. 3: Redirecting if the site isn't safe.

```
if (!sc_site_ssl)
{
sc_redir("http://www.erro_page.com/");
}
```

Note: This macro works only to the web server IIS.

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
	onAfterDelete onAfterInsert onAfterUpdate				onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit				

onExecute	onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onScriptInit	onScriptInit	onApplicationInit onFilterInit	onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onClick onScriptInit onLoadAll onValidate onValidateFailure onValidateSuccess	onApplicationInit onLoad	onLoad	onApplicationInit onScriptInit
-----------	--	-----------------------------------	--------------	-----------------------------------	---	---	-----------------------------	--------	-----------------------------------

sc_sql_injection({My_Field}) or (\$My_Variable)

This macro its used to protect the field/variable against "sql injection".

Macro used for protection against "sql injection" in commands generated by the developer when using the macros: sc_lookup, sc_select or sc_exec_sql.

Ex. 1: Protecting a local variable:
\$field_protect = sc_sql_injection({my_field});

Ex. 2: Protecting an user variable:
\$field_protect = sc_sql_injection(\$my_var);

Note: that all database accesses, generated for the Scriptcase, have protection against "sql injection".

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onFooter onHeader onScriptInit	onClick onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onExecute onLoad	onExecute onLoad	onFooter onHeader onScriptInit onRecord

sc_sql_protect(Value, "Type", "Connection")

This macro will protect the value passed as parameter according with the used database, the "Connection" parameter is optional, if not informed scriptcase will use the default connection.

The "Type" parameter so far can only use the "Date" value.

Ex 01: In this example we will protect the dates passed as parameter so this way it can be correctly interpreted to the ACCESS database.

```
sc_select_where(add) = " AND news_noticias.noticia_data_pub  
BETWEEN ".sc_sql_protect($data_inicial, "date")." AND ".sc_sql_protect($data_final, "date")."";
```

Macro Scope

Blank application	calendar application	Grid application	Search application	Form application	Control Form	ReportPDF application
		onClick onFooter onGroupBy		onAfterDeleteAll onAfterInsertAll onAfterUpdateAll	onLoadAll	

onExecute	onValidate	onScriptInit onNavigate onRecord	onFilterValidate	onBeforeDeleteAll onBeforeInsertAll onBeforeUpdateAll onValidate	onValidate	onRecord
-----------	------------	--	------------------	---	------------	----------

sc_text_style({My_Field}, "Background-Color", "Size", "Color", "Family", "Weight")

This macro allows to modify the text style from the grid field dynamically.

Parameter	Description
Field	Name
Background-Color (optional)	Text cell background Color
Size (optional)	Text font size
Color (optional)	Text Color
Family (optional)	Text family
Weight (optional)	Text weight

Ex. 1:

```
sc_text_style({my_field}, '#33FF99', '15px', '#000000', 'Arial, sans-serif', 'bold');
```

Macro Scope

Grid application

onRecord

sc_time_diff({datetime_01}, "Date_01 Format", {datetime_02}, "Date_02 Format")

sc_time_diff({datetime_01}, "Date_01 Format", {datetime_02}, "Date_02 Format");

This macro calculates the difference between hour, minutes and seconds for two datetime/time values.

The result will be returned in an array, with the dimensions [0], [1] and [2] containing hours, minutes and seconds, respectively.

Parameter	Description
datetime_01	Value or variable for datetime_01
Date_01 Format	Value or variable to specify the format of datetime_01
datetime_02	Value or variable for datetime_02
Date_02 Format	Value or variable to specify the format of datetime_02

Ex. 1: General Format

```
{differences} = sc_time_diff({datetime_01}, "Date_01 Format", {datetime_02}, "Date_02 Format");
```

```
{diff_hours} = {differences[0]};
{diff_minutes} = {differences[1]};
{diff_seconds} = {differences[2]};
```

Ex. 2: Returning positive values

```
{differences} = sc_time_diff("2012-07-25 05:33:45", "yyyy-mm-dd hh:ii:ss", "2012-07-21 15:22:57", "yyyy-mm-dd hh:ii:ss");
```

```
{differences[0]} would be equal to 86 (hours)
{differences[1]} would be equal to 10 (minutes)
{differences[2]} would be equal to 48 (seconds)
```

Ex. 3: Returning negative values

```
{differences} = sc_time_diff("2012-07-21 15:22:57", "yyyy-mm-dd hh:ii:ss", "2012-07-25 05:33:45", "yyyy-mm-dd hh:ii:ss");
```

```
{differences[0]} would be equal to -86 (hours)
{differences[1]} would be equal to -10 (minutes)
{differences[2]} would be equal to -48 (seconds)
```

Ex. 4: Considering only the hours format - Positive return

```
{differences} = sc_time_diff("18:14:55", "hh:ii:ss", "10:55:22", "hh:ii:ss");
```

```
{differences[0]} would be equal to 7 (hours)
{differences[1]} would be equal to 19 (minutes)
{differences[2]} would be equal to 33 (seconds).
```

Ex. 5: Considering only the hours format - Negative return

```
{differences} = sc_time_diff("10:55:22", "hh:ii:ss", "18:14:55", "hh:ii:ss");
```

```
{differences[0]} would be equal to -7 (hours)
{differences[1]} would be equal to -19 (minutes)
{differences[2]} would be equal to -33 (seconds).
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete		onClick		ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll	onBlur onChange onClick			

onExecute	onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onApplicationInit onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onLoad	onApplicationInit onFooter onHeader onScriptInit onRecord
-----------	--	---	--	--	--	---	--	--------	---

sc_trunc_num({My_Field}, Decimal_Number)

This macro is used to truncate numerical values which exceed the decimal amount specified.

Parametro	Descrição
My_Field	Variable with value to format (return in the same variable).
Decimal_Number	Amount of decimals to display.

Ex. 1:
sc_trunc_num({my_value}, 2);

Input value = 1250.235
Output value = 1250.23

Input value = 1250.2
Output value = 1250.20

Input value = 1250
Output value = 1250.00

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onClick onApplicationInit onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onExecute onLoad	onApplicationInit onFooter onHeader onScriptInit onRecord

sc_url_exit(URL)

This macro modifies the application exit URL.

Ex. 1:
sc_url_exit(http://www.scriptcase.net);

Ex. 2:
sc_url_exit(aplx.php);

Macro Scope

calendar application	chart application	Grid application	Search application	Form application	Control Form	ReportPDF application
onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onScriptInit	onScriptInit	onApplicationInit onFilterInit	onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onClick onScriptInit onLoadAll onValidate onValidateFailure onValidateSuccess	onRecord

sc_url_library("Target", "Library Name", "File")

This macro will return the file path, inside a library, to be used on the scriptcase applications. It is necessary to have a library created on Scriptcase to use this Macro. To create a library, go in the "Tools > Library" Menu.

Parameter	Description
Target	Tells you what the scope of the library. Can be "sys" for libraries of <i>Public</i> scope or "prj" for libraries of <i>Project</i> scope.
Library Name	Given name to the library at the time of creation.
File	Absolute path within the library.

Ex. 1 - Importing CSS files from a library:

```
<*link rel="stylesheet" type="text/css" href="<?*php echo sc_url_library('prj', 'bootstrap-3.3.2-dist', 'css/style1.css'); ?>" />
```

Ex. 2 - Importing a JS file from a library:

```
<*script type="text/javascript" src="<?*php echo sc_url_library('prj', 'bootstrap-3.3.2-dist', 'js/bootstrap.js'); ?>" />
```

NOTE: Note that you need to remove the asterisk (*) form the examples above to use it.

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onCalendarApplicationInit onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onScriptInit	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onExecute onLoad

sc_user_logout('variable_name', 'variable_content', 'apl_redir.php', 'target')

This macro is used to log the user out to the system.

If the target is not informed, the default value is "_top".

Ex:

```
sc_user_logout('usr_login', 'admin', 'ctrl_login.php', '_self');
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Form application	Control Form	ReportPDF application
onExecute	onAfterDelete onAfterInsert onAfterUpdate onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onValidate onValidateSuccess	onHeader onScriptInit	onScriptInit	onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onScriptInit onLoad onNavigate onLoadRecord onValidate onValidateSuccess	onScriptInit onLoadAll onValidate onValidateSuccess	onHeader onScriptInit onRecord

sc_warning'on' or 'off'

This macro dynamically activates or deactivates warning messages control.

Warning messages are generated when a reference is made to an inexistent variable, an item of an inexistent array, etc.

Ex. 1: To deactivate the message (do not display).

```
sc_warning = 'off';
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
onExecute	ajaxFieldonBlur OnChange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onScriptInit	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur OnChange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur OnChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onApplicationInit onExecute onLoad	onApplicationInit onFooter onHeader onScriptInit onRecord

sc_webservice("Method", "URL", "Port", "Send Method", "Parameters Array", "Setting's Array", "Timeout", "Return")

This macro is used to communicate with a web service.

Parameter	Description
Method	Informs what method will be used to access the web service. Ex: <i>file_get_contents</i> or <i>curl</i> .
URL	URL used to access the web service.
Port	Port used to access the web service.
Send Method	Defines the request method of the web service. Can be <i>GET</i> ou <i>POST</i> .

Parameters Array	Parameter's array used to access the web service, varying according to the accessed service.
Settings's Array(optional)	Parameter's array for configuration of access method used.
Timeout(optional)	Service's access timeout.
Return(optional)	Web service's return method. When it is set as "True", the return from the web service comes as an array, if it is set as "False" it will return as the original web service format. Default: True

```
$parms = array(
'METHOD' => 'GetBalance',
'RETURNALLCURRENCIES' => 1,
'VERSION' => urlencode(51.0),
'USER' => urlencode({paypal_api_login}),
'PWD' => urlencode({paypal_api_password}),
'SIGNATURE' => urlencode({paypal_api_key})
);
$parms = http_build_query($parms);
```

Ex1 - Using the macro `file_get_contents`:

```
sc_webservice('file_get_contents', $webservice_url, $_SERVER["SERVER_PORT"], "POST", $parms,
array(), 30);
```

Ex2 - Using the macro with `CURL`:

```
$response = sc_webservice("curl", {paypal_type}, 80, "POST", $parms,
array(CURLOPT_RETURNTRANSFER => true, CURLOPT_SSL_VERIFYPEER=>false), 30);
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu
onExecute	ajaxFieldonBlur OnChange OnClick OnFocus onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onFooter onHeader onScriptInit	onScriptInit	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur OnChange OnClick OnFocus onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onBlur onChange onClick onFocus onApplicationInit onClick onScriptInit onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onExecute onLoad

sc_where_current

Variable to reference the application where clause plus the filter selections(if any). Reflects the where clause currently used.

Ex. 1:

```
$save_current_where = {sc_where_current};
```

Macro Scope

chart application	Grid application	ReportPDF application
onFooter onHeader onScriptInit	onClick onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onFooter onHeader onScriptInit onRecord

sc_where_filter

This macro is used to retrieve the content generated by the "filter" according with its selections

This macro is used to retrieve the content generated by the filter, according with its selections. These are the data which will be added to the application original where clause after filter submission.

Ex. 1:

```
$save_current_filter = {sc_where_filter};
```

Macro Scope

chart application	Grid application	ReportPDF application
onFooter onHeader onScriptInit	onClick onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onFooter onHeader onScriptInit onRecord

sc_where_orig

This macro saves the application original where clause. Can be used inside ScriptCase events, to manipulate the data, or to display it anywhere in the grid.

Ex. 1:

```
$save_where = {sc_where_orig};
```

Macro Scope

chart application	Grid application	Search application	ReportPDF application
onFooter onHeader onScriptInit	onClick onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onApplicationInit onFilterInit onFilterRefresh onFilterSave onFilterValidate	onFooter onHeader onScriptInit onRecord

sc_zip_file("File", "Zip")

This macro is used to generate ZIP files from a list of files and/or directories. The file parameter must be one of the following:

- File name
- Directory name
- A variable containing array, which contains a files and/or directories list.
- A variable a file or a directory name.
- A variable containing array, which contains a files and/or directories list.

The zip parameter must contain the name of the zip file generated, or the path to the file.

Ex. 1: Single file compressing.

```
sc_zip_file("/test/sample.htm", "/tmp/test.zip");
```

Ex. 2: Single directory compressing.

```
sc_zip_file("/test", "/tmp/test.zip");
```

Ex. 3: Multiple files and directories in an array.

```
$prep = array();
$prep[] = "/test/sample.htm";
$prep[] = "/test";
sc_zip_file($prep, "/tmp/test.zip");
```

Macro Scope

Blank application	calendar application	chart application	Grid application	Search application	Form application	Control Form	Menu application	Tree menu	ReportPDF application
	onAfterDelete onAfterInsert onAfterUpdate onApplicationInit onBeforeDelete onBeforeInsert	onApplicationInit onFooter		onApplicationInit onFilterInit	onAfterDelete onAfterDeleteAll onAfterInsert onAfterInsertAll onAfterUpdate onAfterUpdateAll onApplicationInit onBeforeDelete onBeforeDeleteAll onBeforeInsert onBeforeInsertAll	onApplicationInit onClick onScriptInit onLoadAll	onApplicationInit	onApplicationInit	onApplicationInit onFooter

onExecute	onBeforeUpdate onScriptInit onLoad onRefresh onValidate onValidateFailure onValidateSuccess	onHeader onScriptInit	onScriptInit	onFilterSave onFilterValidate	onBeforeUpdate onBeforeUpdateAll onClick onScriptInit onLoad onNavigate onLoadRecord onRefresh onValidate onValidateFailure onValidateSuccess	onRefresh onValidate onValidateFailure onValidateSuccess	onExecute onLoad	onExecute onLoad	onHeader onScriptInit onRecord
-----------	---	--------------------------	--------------	----------------------------------	---	---	---------------------	---------------------	--------------------------------------

Authentication Variables

These variables must be defined on applications with PDF export on environments with integrated authentication.

Variables	Description
[sc_pdf_usr]	Authentication User.
[sc_pdf_pw]	Authentication Password

Ex. 1: On a login application, it is possible to set these variables from the info given by the user:

```
[sc_pdf_usr] = {login_usr};
[sc_pdf_pw] = {senha_usr};
```

Macro Scope

Grid application	Form application
onApplicationInit onFooter onGroupBy onScriptInit	onScriptInit

Database Variables

Global variables containing database access values used by main connection.

Variables	Description
sc_glo_server	Server name.
sc_glo_db	Database name.
sc_glo_db_type	Database type (mssql, oracle, mysql, etc).
sc_glo_user	Connected user name.
sc_glo_pass	Database password access.
sc_glo_cript_pass	Encrypted password (S/N).
sc_glo_decimal_db	Used decimal separator (point or comma).

Ex. 1:
\$my_database = [sc_glo_db];

These variables only reference the values without updating it (it is not possible to attribute/modify the values). To define user and the password to connect to the database, use the following variables:

Variables	Description
sc_db_master_usr	User login to be used.
sc_db_master_pass	Password to be used.
sc_db_master_cript	Encrypted password (S/N) (see sc_encode macro).

Important: These variables only take effect during next application execution remaining valid during all the session (unless modified).

Ex. 2: An application type "control" with a form created with information "login" and encrypted "password". These data must be used to connect to the database, during the system applications execution.

```
$temp_pass = sc_encode({password});
[sc_db_master_usr] = {login};
[sc_db_master_pass] = $temp_pass;
[sc_db_master_cript] = "S";
```

Macro Scope

Blank	calendar	chart	Grid	Search	Form	Menu	ReportPDF
-------	----------	-------	------	--------	------	------	-----------

application	application	application	application	application	application	Control Form	application	Tree menu	application
onExecute	ajaxFieldonBlur Onchange OnClick OnFocus	onFooter onHeader onScriptInit	onFooter onGroupBy onHeader onScriptInit onNavigate onRecord	onFilterInit onFilterRefresh onFilterSave onFilterValidate	ajaxFieldonBlur Onchange OnClick OnFocus	onBlur onChange onClick onLoadAll onRefresh onValidate onValidateFailure onValidateSuccess	onApplicationInit onExecute onLoad	onExecute onLoad	onFooter onHeader onScriptInit onRecord

Totalling Variables

During group by and summary processing, Scriptcase generates total variables to each group level.

The routines are defined in event **"onGroupBy"** determining the break levels and available to each record. Totals special variables are then referenced substituting the group name for the key word "groupby", as in:

Variables	Description
{count_ger}	Records total amount.
{sum_parcel}	Will show the total sum for the field "Parcel"
{sum_balance}	Will show the total sum for the field "Balance"
{count_groupby}	Current Group records total.
{sum_groupby_parcel}	Current Group level sum for the field "Parcel".
{sum_groupby_balance}	Current Group level sum for the field "Balance".

Ex. 1: In an application with a group by state and city that totalizes a balance field, in the group totals, is possible to display the average (avg) instead of the balance as follows.

```
{sum_groupby_balance} = {sum_groupby_balance} / {count_groupby};
```

Macro Scope

chart application	Grid application	ReportPDF application
onFooter onHeader	onFooter onGroupBy onRecord	onFooter onHeader onRecord

Totalling Variables (group by)

During the group by processing, Scriptcase generates total variables to each group level. The routines are defined in event **"onGroupBy"** determining the group levels. Totals special variables are then referenced substituting the group name for the key word **"quebra"**, as in:

Variable	Description
{count_ger}	Records total amount.
{sum_parcel}	Field "parcels" total.
{sum_balance}	Field "Balance" total.
{count_quebra}	Current group level records total.
{sum_quebra_parcel}	Current group level field "parcels" total.
{sum_quebra_balance}	Current group level field "Balance" total.

Ex. 1: In an application with a group by state and city and that totalizes a balance field, in the group totals, is possible to display the average (avg) instead of the balance as follows.

```
{sum_quebra_balance} = {sum_quebra_balance} / {count_quebra};
```

Macro Scope

Grid application
onFooter onGroupBy

ajaxFieldonBlur

Aplicação Calendário	Aplicação Formulário	Aplicação Controle
sc_ajax_javascript sc_ajax_message sc_begin_trans sc_calc_dv sc_commit_trans sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_exit sc_error_message sc_exec_sql sc_field_display sc_field_readonly sc_format_num sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_lookup sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_set_global sc_sql_injection sc_time_diff sc_trunc_num sc_url_library sc_warning sc_webservice Variables - Database	sc_ajax_javascript sc_ajax_message sc_begin_trans sc_calc_dv sc_commit_trans sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_exit sc_error_message sc_exec_sql sc_field_display sc_field_readonly sc_format_num sc_format_num_region sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_label sc_lookup sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_set_global sc_sql_injection sc_time_diff sc_trunc_num sc_url_library sc_warning sc_webservice Variables - Databases	sc_ajax_javascript sc_ajax_message sc_begin_trans sc_calc_dv sc_commit_trans sc_concat sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_exit sc_error_message sc_exec_sql sc_field_display sc_field_readonly sc_format_num sc_format_num_region sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_label sc_lookup sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_set_global sc_sql_injection sc_time_diff sc_trunc_num sc_url_library sc_warning sc_webservice Variables - Database

Onchange

Aplicação Calendário	Aplicação Formulário	Aplicação Controle
sc_ajax_javascript sc_ajax_message sc_begin_trans sc_calc_dv sc_commit_trans sc_connection_edit sc_connection_new sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_exit sc_error_message sc_exec_sql sc_field_display sc_field_readonly sc_format_num sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_lookup sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_set_global sc_sql_injection sc_time_diff sc_trunc_num sc_url_library sc_warning sc_webservice Variables - Database	sc_ajax_javascript sc_ajax_message sc_begin_trans sc_calc_dv sc_commit_trans sc_connection_edit sc_connection_new sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_exit sc_error_message sc_exec_sql sc_field_display sc_field_readonly sc_format_num sc_format_num_region sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_label sc_lookup sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_set_global sc_sql_injection sc_time_diff sc_trunc_num sc_url_library sc_warning sc_webservice Variables - Database	sc_ajax_javascript sc_ajax_message sc_begin_trans sc_calc_dv sc_commit_trans sc_concat sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_exit sc_error_message sc_exec_sql sc_field_display sc_field_readonly sc_format_num sc_format_num_region sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_label sc_lookup sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_set_global sc_sql_injection sc_time_diff sc_trunc_num sc_url_library sc_warning sc_webservice Variables - Database

OnClick

Aplicação Calendário	Aplicação Consulta	Aplicação Formulário	Aplicação Controle
sc_ajax_javascript sc_ajax_message sc_begin_trans sc_calc_dv sc_commit_trans sc_concat sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_exit sc_error_message sc_exec_sql sc_field_display sc_field_readonly sc_format_num sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_lookup sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_set_focus sc_set_global sc_sql_injection sc_time_diff sc_trunc_num sc_url_library sc_warning sc_webservice Variables - Database	sc_ajax_refresh sc_begin_trans sc_calc_dv sc_commit_trans sc_concat sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_exec_sql sc_format_num sc_format_num_region sc_label sc_lin_cod_barra_arrecadacao sc_lin_cod_barra_banco sc_lin_digitavel_arrecadacao sc_lin_digitavel_banco sc_lookup sc_mail_send sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_set_global sc_sql_injection sc_sql_protect sc_time_diff sc_trunc_num sc_vl_extenso sc_where_current sc_where_filter sc_where_orig	sc_ajax_javascript sc_ajax_message sc_begin_trans sc_calc_dv sc_commit_trans sc_concat sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_exit sc_error_message sc_exec_sql sc_field_display sc_field_readonly sc_format_num sc_format_num_region sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_label sc_lookup sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_set_focus sc_set_global sc_sql_injection sc_time_diff sc_trunc_num sc_url_library sc_warning sc_webservice Variables - Database	sc_ajax_javascript sc_ajax_message sc_begin_trans sc_calc_dv sc_commit_trans sc_concat sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_exit sc_error_message sc_exec_sql sc_field_display sc_field_readonly sc_format_num sc_format_num_region sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_label sc_lookup sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_set_global sc_sql_injection sc_time_diff sc_trunc_num sc_url_library sc_warning sc_webservice Variables - Database

OnFocus

Aplicação Calendário	Aplicação Formulário	Aplicação Controle
sc_ajax_javascript sc_ajax_message sc_begin_trans sc_calc_dv sc_commit_trans sc_concat sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_exit sc_error_message sc_exec_sql sc_field_display sc_field_readonly sc_format_num sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_lookup sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_set_global sc_sql_injection sc_time_diff sc_trunc_num sc_url_library sc_warning sc_webservice Variables - Database	sc_ajax_javascript sc_ajax_message sc_begin_trans sc_calc_dv sc_commit_trans sc_concat sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_exit sc_error_message sc_exec_sql sc_field_display sc_field_readonly sc_format_num sc_format_num_region sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_label sc_lookup sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_set_global sc_sql_injection sc_time_diff sc_trunc_num sc_url_library sc_warning sc_webservice Variables - Database	sc_ajax_javascript sc_ajax_message sc_begin_trans sc_calc_dv sc_commit_trans sc_concat sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_exit sc_error_message sc_exec_sql sc_field_display sc_field_readonly sc_format_num sc_format_num_region sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_label sc_lookup sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_set_global sc_sql_injection sc_time_diff sc_trunc_num sc_url_library sc_warning sc_webservice

onAfterDelete

Aplicação Calendário

sc_ajax_javascript
 sc_begin_trans
 sc_btn_display
 sc_calc_dv
 sc_commit_trans
 sc_concat
 sc_connection_edit
 sc_connection_new
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_date_empty
 sc_decode
 sc_encode
 sc_error_delete
 sc_error_exit
 sc_error_message
 sc_exec_sql
 sc_field_display
 sc_field_readonly
 sc_format_num
 sc_get_language
 sc_get_regional
 sc_get_theme
 sc_include
 sc_include_library
 sc_lookup
 sc_mail_send
 sc_master_value
 sc_redir
 sc_reset_connection_edit
 sc_reset_connection_new
 sc_reset_global
 sc_rollback_trans
 sc_select
 sc_set_fetchmode
 sc_set_global
 sc_site_ssl
 sc_sql_injection
 sc_time_diff
 sc_trunc_num
 sc_url_exit
 sc_url_library
 sc_user_logout
 sc_warning
 sc_webservice
 sc_zip_file

Aplicação Formulário

sc_ajax_javascript
 sc_begin_trans
 sc_btn_display
 sc_calc_dv
 sc_commit_trans
 sc_concat
 sc_connection_edit
 sc_connection_new
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_date_empty
 sc_decode
 sc_encode
 sc_error_delete
 sc_error_exit
 sc_error_message
 sc_exec_sql
 sc_field_display
 sc_field_readonly
 sc_format_num
 sc_format_num_region
 sc_get_language
 sc_get_regional
 sc_get_theme
 sc_include
 sc_include_library
 sc_lookup
 sc_mail_send
 sc_master_value
 sc_redir
 sc_reset_connection_edit
 sc_reset_connection_new
 sc_reset_global
 sc_rollback_trans
 sc_select
 sc_set_fetchmode
 sc_set_global
 sc_site_ssl
 sc_sql_injection
 sc_time_diff
 sc_trunc_num
 sc_url_exit
 sc_url_library
 sc_user_logout
 sc_warning
 sc_webservice
 sc_zip_file

onAfterDeleteAll

Aplicação Formulário

sc_ajax_javascript
 sc_begin_trans
 sc_block_display
 sc_btn_display
 sc_calc_dv
 sc_commit_trans
 sc_concat
 sc_connection_edit
 sc_connection_new
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_date_empty
 sc_decode
 sc_encode
 sc_error_exit
 sc_error_message
 sc_exec_sql
 sc_field_display
 sc_field_readonly
 sc_format_num
 sc_format_num_region
 sc_get_language
 sc_get_regional
 sc_get_theme
 sc_include
 sc_include_library
 sc_lookup
 sc_mail_send
 sc_master_value
 sc_redir
 sc_reset_connection_edit
 sc_reset_connection_new
 sc_reset_global
 sc_rollback_trans
 sc_select
 sc_set_fetchmode
 sc_set_global
 sc_site_ssl
 sc_sql_injection
 sc_sql_protect
 sc_time_diff
 sc_trunc_num
 sc_url_exit

sc_url_library
 sc_user_logout
 sc_warning
 sc_webservice
 sc_zip_file

Aplicação Formulário

onAfterInsert

Aplicação Calendário

sc_ajax_javascript
 sc_begin_trans
 sc_block_display
 sc_btn_display
 sc_calc_dv
 sc_commit_trans
 sc_concat
 sc_connection_edit
 sc_connection_new
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_date_empty
 sc_decode
 sc_encode
 sc_error_exit
 sc_error_insert
 sc_error_message
 sc_exec_sql
 sc_field_display
 sc_field_readonly
 sc_format_num
 sc_get_language
 sc_get_regional
 sc_get_theme
 sc_include
 sc_include_library
 sc_lookup
 sc_mail_send
 sc_master_value
 sc_redir
 sc_reset_connection_edit
 sc_reset_connection_new
 sc_reset_global
 sc_rollback_trans
 sc_select
 sc_set_fetchmode
 sc_set_global
 sc_site_ssl
 sc_sql_injection
 sc_time_diff
 sc_trunc_num
 sc_url_exit
 sc_url_library
 sc_user_logout
 sc_warning
 sc_webservice
 sc_zip_file

Aplicação Formulário

sc_ajax_javascript
 sc_begin_trans
 sc_block_display
 sc_btn_display
 sc_calc_dv
 sc_commit_trans
 sc_concat
 sc_connection_edit
 sc_connection_new
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_date_empty
 sc_decode
 sc_encode
 sc_error_exit
 sc_error_insert
 sc_error_message
 sc_exec_sql
 sc_field_display
 sc_field_readonly
 sc_format_num
 sc_format_num_region
 sc_get_language
 sc_get_regional
 sc_get_theme
 sc_include
 sc_include_library
 sc_lookup
 sc_mail_send
 sc_master_value
 sc_redir
 sc_reset_connection_edit
 sc_reset_connection_new
 sc_reset_global
 sc_rollback_trans
 sc_select
 sc_set_fetchmode
 sc_set_global
 sc_site_ssl
 sc_sql_injection
 sc_time_diff
 sc_trunc_num
 sc_url_exit
 sc_url_library
 sc_user_logout
 sc_warning
 sc_webservice
 sc_zip_file

onAfterInsertAll

Aplicação Formulário

sc_ajax_javascript
 sc_begin_trans
 sc_block_display
 sc_btn_display
 sc_calc_dv
 sc_commit_trans
 sc_concat
 sc_connection_edit
 sc_connection_new
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_date_empty
 sc_decode
 sc_encode
 sc_error_exit
 sc_error_message
 sc_exec_sql
 sc_field_display
 sc_field_readonly
 sc_format_num
 sc_format_num_region
 sc_get_language
 sc_get_regional
 sc_get_theme
 sc_include
 sc_include_library
 sc_lookup
 sc_mail_send
 sc_master_value
 sc_redir
 sc_reset_connection_edit
 sc_reset_connection_new
 sc_reset_global
 sc_rollback_trans
 sc_select
 sc_set_fetchmode

```

sc_set_global
sc_site_ssl
sc_sql_injection
sc_sql_protect
sc_time_diff
sc_trunc_num
sc_url_exit
Aplicação Formulário
sc_url_library
sc_user_logout
sc_warning
sc_webservice
sc_zip_file

```

onAfterUpdate

Aplicação Calendário

```

sc_ajax_javascript
sc_begin_trans
sc_block_display
sc_btn_display
sc_calc_dv
sc_commit_trans
sc_concat
sc_connection_edit
sc_connection_new
sc_date
sc_date_conv
sc_date_dif
sc_date_dif_2
sc_date_empty
sc_decode
sc_encode
sc_error_exit
sc_error_message
sc_error_update
sc_exec_sql
sc_field_display
sc_field_readonly
sc_format_num
sc_get_language
sc_get_regional
sc_get_theme
sc_include
sc_include_library
sc_lookup
sc_mail_send
sc_master_value
sc_redir
sc_reset_connection_edit
sc_reset_connection_new
sc_reset_global
sc_rollback_trans
sc_select
sc_set_fetchmode
sc_set_global
sc_site_ssl
sc_sql_injection
sc_time_diff
sc_trunc_num
sc_url_exit
sc_url_library
sc_user_logout
sc_warning
sc_webservice
sc_zip_file

```

Aplicação Formulário

```

sc_ajax_javascript
sc_begin_trans
sc_block_display
sc_btn_display
sc_calc_dv
sc_commit_trans
sc_concat
sc_connection_edit
sc_connection_new
sc_date
sc_date_conv
sc_date_dif
sc_date_dif_2
sc_date_empty
sc_decode
sc_encode
sc_error_exit
sc_error_message
sc_error_update
sc_exec_sql
sc_field_display
sc_field_readonly
sc_format_num
sc_format_num_region
sc_get_language
sc_get_regional
sc_get_theme
sc_include
sc_include_library
sc_lookup
sc_mail_send
sc_master_value
sc_redir
sc_reset_connection_edit
sc_reset_connection_new
sc_reset_global
sc_rollback_trans
sc_select
sc_set_fetchmode
sc_set_global
sc_site_ssl
sc_sql_injection
sc_time_diff
sc_trunc_num
sc_url_exit
sc_url_library
sc_user_logout
sc_warning
sc_webservice
sc_zip_file

```

onAfterUpdateAll

Aplicação Formulário

```

sc_ajax_javascript
sc_begin_trans
sc_block_display
sc_btn_display
sc_calc_dv
sc_commit_trans
sc_concat
sc_connection_edit
sc_connection_new
sc_date
sc_date_conv
sc_date_dif
sc_date_dif_2
sc_date_empty
sc_decode
sc_encode
sc_error_exit
sc_error_message
sc_exec_sql
sc_field_display
sc_field_readonly
sc_format_num
sc_format_num_region
sc_get_language
sc_get_regional
sc_get_theme
sc_include
sc_include_library
sc_lookup
sc_mail_send
sc_master_value

```

sc_redir
 sc_reset_connection_edit
 sc_reset_connection_new
 sc_reset_global
 sc_rollback_trans
 sc_select
 sc_set_fetchmode
 Aplicação Formulário
 sc_set_global
 sc_site_ssl
 sc_sql_injection
 sc_sql_protect
 sc_time_diff
 sc_trunc_num
 sc_url_exit
 sc_url_library
 sc_user_logout
 sc_warning
 sc_webservice
 sc_zip_file

onApplicationInit

Aplicação Calendário	Aplicação Gráfico	Aplicação Consulta	Aplicação Filtro	Aplicação Formulário	Aplicação Controle	Aplicação Menu	Aplicação Menu árvore	Aplicação Report PDF
sc_apl_conf sc_apl_status sc_calc_dv sc_connection_edit sc_connection_new sc_date sc_date_dif sc_date_dif_2 sc_decode sc_encode sc_format_num sc_get_language sc_get_regional sc_get_theme sc_image sc_include sc_include_library sc_language sc_ldap_login sc_ldap_logout sc_ldap_search sc_master_value sc_redir sc_reset_apl_conf sc_reset_apl_status sc_reset_apl_status sc_reset_change_connection sc_reset_connection_edit sc_reset_connection_new n sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_set_groupby_rule sc_set_global sc_set_language sc_set_regional sc_set_theme sc_site_ssl sc_time_diff sc_trunc_num sc_url_exit sc_url_library sc_warning sc_webservice sc_zip_file	sc_apl_conf sc_apl_status sc_calc_dv sc_connection_edit sc_connection_new sc_date sc_date_dif sc_date_dif_2 sc_decode sc_encode sc_field_display sc_format_num sc_get_language sc_get_regional sc_get_theme sc_image sc_hide_groupby_rule sc_image sc_include sc_include_library sc_language sc_ldap_login sc_ldap_logout sc_ldap_search sc_master_value sc_redir sc_reset_apl_conf sc_reset_apl_status sc_reset_change_connection sc_reset_connection_edit sc_reset_connection_new sc_set_groupby_rule sc_set_global sc_set_language sc_set_regional sc_set_theme sc_site_ssl sc_time_diff sc_trunc_num Variables - Authentication	sc_apl_conf sc_apl_status sc_calc_dv sc_change_connection sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_field_init_off sc_format_num sc_get_language sc_get_regional sc_format_num_region sc_redir sc_reset_apl_conf sc_reset_apl_status sc_reset_change_connection sc_reset_global sc_set_global sc_set_groupby_rule sc_set_language sc_set_regional sc_set_theme sc_time_diff sc_trunc_num Variables - Authentication	sc_apl_conf sc_apl_status sc_calc_dv sc_connection_edit sc_connection_new sc_date sc_date_dif sc_date_dif_2 sc_decode sc_encode sc_format_num sc_format_num_region sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_language sc_reset_apl_conf sc_reset_apl_status sc_reset_connection_edit sc_reset_connection_new sc_set_global sc_set_language sc_set_regional sc_set_theme sc_time_diff sc_trunc_num sc_url_exit sc_url_library sc_warning sc_webservice sc_zip_file	sc_apl_conf sc_apl_status sc_calc_dv sc_connection_edit sc_connection_new sc_date sc_date_dif sc_date_dif_2 sc_decode sc_encode sc_format_num sc_format_num_region sc_get_language sc_get_regional sc_get_theme sc_image sc_include sc_include_library sc_language sc_ldap_login sc_ldap_logout sc_ldap_search sc_master_value sc_redir sc_reset_apl_conf sc_reset_apl_status sc_reset_apl_status sc_reset_change_connection sc_reset_connection_edit sc_reset_connection_new sc_reset_connection_new sc_reset_global sc_set_global sc_set_language sc_set_regional sc_set_theme sc_time_diff sc_trunc_num sc_url_exit sc_url_library sc_warning sc_webservice sc_zip_file	sc_apl_conf sc_apl_status sc_calc_dv sc_connection_edit sc_connection_new sc_date sc_date_dif sc_date_dif_2 sc_decode sc_encode sc_format_num sc_format_num_region sc_get_language sc_get_regional sc_get_theme sc_image sc_include sc_include_library sc_language sc_ldap_login sc_ldap_logout sc_ldap_search sc_redir sc_reset_apl_conf sc_reset_apl_status sc_reset_apl_status sc_reset_change_connection sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_set_global sc_set_language sc_set_regional sc_set_theme sc_time_diff sc_trunc_num sc_url_exit sc_url_library sc_warning sc_webservice sc_zip_file	sc_apl_conf sc_apl_status sc_appmenu_add_item sc_appmenu_create sc_appmenu_create sc_appmenu_exist_item sc_appmenu_remove_item sc_appmenu_remove_item sc_appmenu_reset sc_appmenu_reset sc_appmenu_update_item sc_appmenu_update_item sc_begin_trans sc_btn_disable sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_decode sc_encode sc_exec_sql sc_format_num sc_format_num_region sc_get_language sc_get_regional sc_get_theme sc_image sc_include sc_include_library sc_language sc_ldap_login sc_ldap_logout sc_ldap_search sc_mail_send sc_mail_send sc_menu_delete sc_menu_delete sc_menu_disable sc_menu_force_mobile sc_menu_item sc_menu_item sc_reset_apl_conf sc_reset_apl_status sc_reset_apl_status sc_reset_change_connection sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_set_global sc_set_language sc_set_regional sc_set_theme sc_time_diff sc_trunc_num sc_url_exit sc_url_library sc_warning sc_webservice sc_zip_file Variables - Database	sc_apl_conf sc_apl_status sc_appmenu_add_item sc_appmenu_create sc_appmenu_create sc_appmenu_exist_item sc_appmenu_remove_item sc_appmenu_remove_item sc_appmenu_reset sc_appmenu_reset sc_appmenu_update_item sc_appmenu_update_item sc_begin_trans sc_commit_trans sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_decode sc_encode sc_exec_sql sc_format_num sc_format_num_region sc_get_language sc_get_regional sc_get_theme sc_image sc_include sc_include_library sc_language sc_ldap_login sc_ldap_logout sc_ldap_search sc_mail_send sc_mail_send sc_menu_delete sc_menu_delete sc_menu_disable sc_menu_force_mobile sc_menu_item sc_menu_item sc_reset_apl_conf sc_reset_apl_status sc_reset_apl_status sc_reset_change_connection sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_set_global sc_set_language sc_set_regional sc_set_theme sc_time_diff sc_trunc_num sc_url_exit sc_url_library sc_warning sc_webservice sc_zip_file	sc_apl_conf sc_apl_status sc_calc_dv sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_decode sc_encode sc_format_num sc_get_theme sc_image sc_include sc_include_library sc_language sc_reset_apl_conf sc_reset_apl_status sc_reset_apl_status sc_reset_change_connection sc_reset_connection_edit sc_reset_connection_new sc_set_global sc_set_language sc_set_regional sc_set_theme sc_time_diff sc_trunc_num sc_url_exit sc_url_library sc_warning sc_webservice sc_zip_file

onBeforeDelete

Aplicação Calendário	Aplicação Formulário
sc_begin_trans sc_calc_dv sc_commit_trans sc_concat sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_continue sc_error_exit sc_error_message sc_exec_sql	sc_begin_trans sc_calc_dv sc_commit_trans sc_concat sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_continue sc_error_exit sc_error_message sc_exec_sql

sc_format_num	sc_format_num
sc_get_language	sc_format_num_region
sc_get_regional	sc_get_language
sc_get_theme	sc_get_regional
sc_include	sc_get_theme
sc_include_library	sc_include
sc_lookup	sc_include_library
Aplicação Calendário	Aplicação Formulário
sc_mail_send	sc_lookup
sc_master_value	sc_mail_send
sc_redir	sc_master_value
sc_reset_connection_edit	sc_redir
sc_reset_connection_new	sc_reset_connection_edit
sc_reset_global	sc_reset_connection_new
sc_rollback_trans	sc_reset_global
sc_select	sc_rollback_trans
sc_set_fetchmode	sc_select
sc_set_global	sc_set_fetchmode
sc_site_ssl	sc_set_global
sc_sql_injection	sc_site_ssl
sc_time_diff	sc_sql_injection
sc_trunc_num	sc_time_diff
sc_url_exit	sc_trunc_num
sc_url_library	sc_url_exit
sc_user_logout	sc_url_library
sc_warning	sc_user_logout
sc_webservice	sc_warning
sc_zip_file	sc_webservice
	sc_zip_file

onBeforeDeleteAll

Aplicação Formulário

sc_begin_trans
sc_calc_dv
sc_commit_trans
sc_concat
sc_connection_edit
sc_connection_new
sc_date
sc_date_conv
sc_date_dif
sc_date_dif_2
sc_date_empty
sc_decode
sc_encode
sc_error_exit
sc_error_message
sc_exec_sql
sc_format_num
sc_format_num_region
sc_get_language
sc_get_regional
sc_get_theme
sc_include
sc_include_library
sc_lookup
sc_mail_send
sc_redir
sc_reset_connection_edit
sc_reset_connection_new
sc_reset_global
sc_rollback_trans
sc_select
sc_set_fetchmode
sc_set_global
sc_site_ssl
sc_sql_injection
sc_sql_protect
sc_time_diff
sc_trunc_num
sc_url_exit
sc_url_library
sc_user_logout
sc_warning
sc_webservice
sc_zip_file

onBeforeInsert

Aplicação Calendário

sc_begin_trans
sc_calc_dv
sc_commit_trans
sc_concat
sc_connection_edit
sc_connection_new
sc_date
sc_date_conv
sc_date_dif
sc_date_dif_2
sc_date_empty
sc_decode
sc_encode
sc_error_continue
sc_error_exit
sc_error_message
sc_exec_sql
sc_format_num
sc_get_language
sc_get_regional
sc_get_theme

Aplicação Formulário

sc_begin_trans
sc_calc_dv
sc_commit_trans
sc_concat
sc_connection_edit
sc_connection_new
sc_date
sc_date_conv
sc_date_dif
sc_date_dif_2
sc_date_empty
sc_decode
sc_encode
sc_error_continue
sc_error_exit
sc_error_message
sc_exec_sql
sc_format_num
sc_format_num_region
sc_get_language
sc_get_regional

```

sc_include
sc_include_library
sc_lin_cod_barra_arrecadacao
sc_lin_cod_barra_banco
sc_lin_digitavel_arrecadacao
sc_lin_digitavel_banco
sc_lookup
Aplicação Calendário
sc_mail_send
sc_master_value
sc_redir
sc_reset_connection_edit
sc_reset_connection_new
sc_reset_global
sc_rollback_trans
sc_select
sc_set_fetchmode
sc_set_global
sc_site_ssl
sc_sql_injection
sc_time_diff
sc_trunc_num
sc_url_exit
sc_url_library
sc_user_logout
sc_warning
sc_webservice
sc_zip_file

```

```

sc_get_theme
sc_include
sc_include_library
sc_lin_cod_barra_arrecadacao
sc_lin_cod_barra_banco
sc_lin_digitavel_arrecadacao
sc_lin_digitavel_banco
Aplicação Formulário
sc_lookup
sc_mail_send
sc_master_value
sc_redir
sc_reset_connection_edit
sc_reset_connection_new
sc_reset_global
sc_rollback_trans
sc_select
sc_set_fetchmode
sc_set_global
sc_site_ssl
sc_sql_injection
sc_time_diff
sc_trunc_num
sc_url_exit
sc_url_library
sc_user_logout
sc_warning
sc_webservice
sc_zip_file

```

onBeforeInsertAll

Aplicação Formulário

```

sc_begin_trans
sc_calc_dv
sc_commit_trans
sc_concat
sc_connection_edit
sc_connection_new
sc_date
sc_date_conv
sc_date_dif
sc_date_dif_2
sc_date_empty
sc_decode
sc_encode
sc_error_exit
sc_error_message
sc_exec_sql
sc_field_display
sc_field_readonly
sc_format_num
sc_format_num_region
sc_get_language
sc_get_regional
sc_get_theme
sc_include
sc_include_library
sc_lookup
sc_mail_send
sc_redir
sc_reset_connection_edit
sc_reset_connection_new
sc_reset_global
sc_rollback_trans
sc_select
sc_set_fetchmode
sc_set_global
sc_site_ssl
sc_sql_injection
sc_sql_protect
sc_time_diff
sc_trunc_num
sc_url_exit
sc_url_library
sc_user_logout
sc_warning
sc_webservice
sc_zip_file

```

onBeforeUpdate

Aplicação Calendário

```

sc_begin_trans
sc_calc_dv
sc_changed
sc_commit_trans
sc_concat
sc_connection_edit
sc_connection_new
sc_date
sc_date_conv
sc_date_dif
sc_date_dif_2
sc_date_empty
sc_decode
sc_encode
sc_error_continue
sc_error_exit
sc_error_message
sc_exec_sql
sc_format_num

```

Aplicação Formulário

```

sc_begin_trans
sc_calc_dv
sc_changed
sc_commit_trans
sc_concat
sc_connection_edit
sc_connection_new
sc_date
sc_date_conv
sc_date_dif
sc_date_dif_2
sc_date_empty
sc_decode
sc_encode
sc_error_continue
sc_error_exit
sc_error_message
sc_exec_sql
sc_format_num

```

sc_get_language	sc_format_num_region
sc_get_regional	sc_get_language
sc_get_theme	sc_get_regional
sc_include	sc_get_theme
sc_include_library	sc_include
sc_lin_cod_barra_arrecadacao	sc_include_library
sc_lin_cod_barra_banco	sc_lin_cod_barra_arrecadacao
Aplicação Calendário	Aplicação Formulário
sc_lin_digitavel_arrecadacao	sc_lin_cod_barra_banco
sc_lin_digitavel_banco	sc_lin_digitavel_arrecadacao
sc_lookup	sc_lin_digitavel_banco
sc_mail_send	sc_lookup
sc_master_value	sc_mail_send
sc_redir	sc_master_value
sc_reset_connection_edit	sc_redir
sc_reset_connection_new	sc_reset_connection_edit
sc_reset_global	sc_reset_connection_new
sc_rollback_trans	sc_reset_global
sc_select	sc_rollback_trans
sc_set_fetchmode	sc_select
sc_set_global	sc_set_fetchmode
sc_site_ssl	sc_set_global
sc_sql_injection	sc_site_ssl
sc_time_diff	sc_sql_injection
sc_trunc_num	sc_time_diff
sc_url_exit	sc_trunc_num
sc_url_library	sc_url_exit
sc_user_logout	sc_url_library
sc_warning	sc_user_logout
sc_webservice	sc_warning
sc_zip_file	sc_webservice
	sc_zip_file

onBeforeUpdateAll

Aplicação Formulário

```

sc_begin_trans
sc_calc_dv
sc_commit_trans
sc_concat
sc_connection_edit
sc_connection_new
sc_date
sc_date_conv
sc_date_dif
sc_date_dif_2
sc_date_empty
sc_decode
sc_encode
sc_error_exit
sc_error_message
sc_exec_sql
sc_field_display
sc_field_readonly
sc_format_num
sc_format_num_region
sc_get_language
sc_get_regional
sc_get_theme
sc_include
sc_include_library
sc_lookup
sc_mail_send
sc_redir
sc_reset_connection_edit
sc_reset_connection_new
sc_reset_global
sc_rollback_trans
sc_select
sc_set_fetchmode
sc_set_global
sc_site_ssl
sc_sql_injection
sc_sql_protect
sc_time_diff
sc_trunc_num
sc_url_exit
sc_url_library
sc_user_logout
sc_warning
sc_webservice
sc_zip_file

```

onExecute

Aplicação Blank

```

sc_apl_conf
sc_apl_status
sc_begin_trans
sc_calc_dv
sc_change_connection
sc_commit_trans
sc_concat
sc_connection_edit
sc_connection_new
sc_date
sc_date_conv
sc_date_dif
sc_date_dif_2
sc_date_empty
sc_decode
sc_encode

```

Aplicação Menu

```

sc_apl_conf
sc_begin_trans
sc_calc_dv
sc_commit_trans
sc_concat
sc_connection_edit
sc_connection_new
sc_date
sc_date_dif
sc_date_dif_2
sc_decode
sc_encode
sc_exec_sql
sc_format_num
sc_format_num_region
sc_get_language

```

Aplicação Menu árvore

```

sc_apl_conf
sc_begin_trans
sc_calc_dv
sc_commit_trans
sc_concat
sc_connection_edit
sc_connection_new
sc_date
sc_date_dif
sc_date_dif_2
sc_decode
sc_encode
sc_exec_sql
sc_format_num
sc_format_num_region
sc_get_language

```

sc_exec_sql	sc_get_regional	sc_get_regional
sc_format_num	sc_get_theme	sc_get_theme
sc_format_num_region	sc_include	sc_include
sc_get_language	sc_include_library	sc_include_library
sc_get_regional	sc_lookup	sc_lookup
sc_get_theme	sc_mail_send	sc_mail_send
sc_image	sc_menu_item	sc_menu_item
Aplicação Blank	Aplicação Menu	Aplicação Menu árvore
sc_include	sc_redir	sc_redir
sc_include_lib	sc_reset_connection_edit	sc_reset_connection_edit
sc_include_library	sc_reset_connection_new	sc_reset_connection_new
sc_language	sc_reset_global	sc_rollback_trans
sc_lin_cod_barra_arrecadacao	sc_rollback_trans	sc_script_name
sc_lin_cod_barra_banco	sc_script_name	sc_select
sc_lin_digitavel_arrecadacao	sc_select	sc_set_fetchmode
sc_lin_digitavel_banco	sc_set_fetchmode	sc_set_global
sc_log_add	sc_set_global	sc_sql_injection
sc_log_split	sc_sql_injection	sc_trunc_num
sc_lookup	sc_time_diff	sc_url_library
sc_mail_send	sc_trunc_num	sc_warning
sc_make_link	sc_url_library	sc_webservice
sc_redir	sc_warning	sc_zip_file
sc_reset_apl_conf	sc_webservice	Variables - Database
sc_reset_apl_status	sc_zip_file	
sc_reset_change_connection	Variables - Database	
sc_reset_connection_edit		
sc_reset_connection_new		
sc_reset_global		
sc_rollback_trans		
sc_select		
sc_set_fetchmode		
sc_set_global		
sc_set_language		
sc_set_regional		
sc_set_theme		
sc_site_ssl		
sc_sql_injection		
sc_sql_protect		
sc_time_diff		
sc_trunc_num		
sc_url_library		
sc_user_logout		
sc_warning		
sc_webservice		
sc_zip_file		
Variables - Database		

onFilterInit

Aplicação Filtro

sc_apl_conf
 sc_apl_status
 sc_begin_trans
 sc_calc_dv
 sc_commit_trans
 sc_concat
 sc_connection_edit
 sc_connection_new
 sc_date
 sc_date_conv
 sc_date_diff
 sc_date_diff_2
 sc_decode
 sc_encode
 sc_exec_sql
 sc_format_num
 sc_format_num_region
 sc_get_language
 sc_get_regional
 sc_get_theme
 sc_image
 sc_include
 sc_include_library
 sc_language
 sc_lookup
 sc_mail_send
 sc_redir
 sc_reset_apl_conf
 sc_reset_apl_status
 sc_reset_connection_edit
 sc_reset_connection_new
 sc_reset_global
 sc_rollback_trans
 sc_select
 sc_set_fetchmode
 sc_set_global
 sc_set_language
 sc_set_regional
 sc_set_theme
 sc_site_ssl
 sc_sql_injection
 sc_time_diff
 sc_trunc_num
 sc_url_exit
 sc_url_library
 sc_warning
 sc_webservice
 sc_zip_file
 Variables - Database

onFilterRefresh

Aplicação Filtro

sc_begin_trans
sc_calc_dv
sc_commit_trans
sc_concat
sc_connection_edit
sc_connection_new
sc_date
sc_date_conv
sc_date_dif
sc_date_dif_2
sc_date_empty
sc_decode
sc_encode
sc_exec_sql
sc_format_num
sc_format_num_region
sc_get_language
sc_get_regional
sc_get_theme
sc_include
sc_include_library
sc_lookup
sc_reset_connection_edit
sc_reset_connection_new
sc_reset_global
sc_rollback_trans
sc_select
sc_set_fetchmode
sc_set_global
sc_sql_injection
sc_time_diff
sc_trunc_num
sc_url_library
sc_warning
sc_webservice
Variables - Database

onFilterSave

Aplicação Filtro

sc_begin_trans
sc_calc_dv
sc_commit_trans
sc_concat
sc_connection_edit
sc_connection_new
sc_date
sc_date_conv
sc_date_dif
sc_date_dif_2
sc_decode
sc_encode
sc_exec_sql
sc_format_num
sc_format_num_region
sc_get_language
sc_get_regional
sc_get_theme
sc_include
sc_include_library
sc_lookup
sc_mail_send
sc_reset_connection_edit
sc_reset_connection_new
sc_reset_global
sc_rollback_trans
sc_select
sc_set_fetchmode
sc_set_global
sc_sql_injection
sc_time_diff
sc_trunc_num
sc_url_library
sc_warning
sc_webservice
sc_zip_file
Variables - Database

onFilterValidate

Aplicação Filtro

sc_begin_trans
sc_calc_dv
sc_commit_trans
sc_concat
sc_connection_edit
sc_connection_new
sc_date
sc_date_conv
sc_date_dif
sc_date_dif_2
sc_date_empty

sc_decode
 sc_encode
 sc_error_exit
 sc_error_message
 sc_exec_sql
 sc_format_num
 sc_format_num_region
 Aplicação Filtro
 sc_get_language
 sc_get_regional
 sc_get_theme
 sc_include
 sc_include_library
 sc_lookup
 sc_mail_send
 sc_redir
 sc_reset_connection_edit
 sc_reset_connection_new
 sc_reset_global
 sc_rollback_trans
 sc_select
 sc_set_fetchmode
 sc_set_global
 sc_set_language
 sc_set_regional
 sc_set_theme
 sc_sql_injection
 sc_sql_protect
 sc_time_diff
 sc_trunc_num
 sc_url_library
 sc_warning
 sc_webservice
 sc_zip_file
 Variables - Database

onFooter

Aplicação Gráfico

sc_begin_trans
 sc_calc_dv
 sc_commit_trans
 sc_concat
 sc_connection_edit
 sc_connection_new
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_decode
 sc_encode
 sc_exec_sql
 sc_format_num
 sc_get_groupby_rule
 sc_get_language
 sc_get_regional
 sc_get_theme
 sc_include_library
 sc_lin_cod_barra_arrecadacao
 sc_lin_cod_barra_banco
 sc_lin_digitavel_arrecadacao
 sc_lin_digitavel_banco
 sc_lookup
 sc_mail_send
 sc_make_link
 sc_reset_connection_edit
 sc_reset_connection_new
 sc_reset_global
 sc_select
 sc_set_fetchmode
 sc_set_global
 sc_sql_injection
 sc_time_diff
 sc_trunc_num
 sc_url_library
 sc_warning
 sc_webservice
 sc_where_current
 sc_where_filter
 sc_where_orig
 sc_zip_file
 Variables - Database
 Variables - Totalling

Aplicação Consulta

sc_begin_trans
 sc_calc_dv
 sc_commit_trans
 sc_concat
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_date_empty
 sc_format_num
 sc_format_num_region
 sc_get_language
 sc_get_regional
 sc_get_theme
 sc_image
 sc_lin_cod_barra_arrecadacao
 sc_lin_cod_barra_banco
 sc_lin_digitavel_arrecadacao
 sc_lin_digitavel_banco
 sc_lookup
 sc_rollback_trans
 sc_select
 sc_set_fetchmode
 sc_set_global
 sc_sql_injection
 sc_sql_protect
 sc_time_diff
 sc_trunc_num
 sc_vl_extenso
 sc_where_current
 sc_where_filter
 sc_where_orig
 Variables - Authentication
 Variables - Database
 Variables - Totalling
 Variables - Totalling (group by)

Aplicação ReportPDF

sc_begin_trans
 sc_calc_dv
 sc_commit_trans
 sc_concat
 sc_connection_edit
 sc_connection_new
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_decode
 sc_encode
 sc_exec_sql
 sc_format_num
 sc_get_language
 sc_get_theme
 sc_lin_cod_barra_arrecadacao
 sc_lin_cod_barra_banco
 sc_lin_digitavel_arrecadacao
 sc_lin_digitavel_banco
 sc_lookup
 sc_mail_send
 sc_reset_connection_edit
 sc_reset_connection_new
 sc_reset_global
 sc_select
 sc_set_fetchmode
 sc_set_global
 sc_sql_injection
 sc_time_diff
 sc_trunc_num
 sc_warning
 sc_where_current
 sc_where_filter
 sc_where_orig
 sc_zip_file
 Variables - Database
 Variables - Totalling

onGroupBy

Aplicação Consulta

sc_begin_trans
 sc_calc_dv
 sc_commit_trans
 sc_concat
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_date_empty
 sc_exec_sql
 sc_format_num_region
 sc_groupby_label
 sc_lookup
 sc_reset_global
 sc_rollback_trans

sc_select
 sc_set_fetchmode
 sc_set_global
 sc_sql_injection
 sc_sql_protect
 sc_time_diff
 sc_trunc_num
 Aplicação Consulta
 sc_where_current
 sc_where_filter
 sc_where_orig
 Variables - Authentication
 Variables - Database
 Variables - Totalling
 Variables - Totalling (group by)

onHeader

Aplicação Gráfico

sc_begin_trans
 sc_calc_dv
 sc_change_connection
 sc_commit_trans
 sc_concat
 sc_connection_edit
 sc_connection_new
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_date_empty
 sc_decode
 sc_encode
 sc_exec_sql
 sc_field_display
 sc_format_num
 sc_get_groupby_rule
 sc_get_language
 sc_get_regional
 sc_get_theme
 sc_include
 sc_include_library
 sc_lin_cod_barra_arrecadacao
 sc_lin_cod_barra_banco
 sc_lin_digitavel_arrecadacao
 sc_lin_digitavel_banco
 sc_lookup
 sc_mail_send
 sc_make_link
 sc_reset_connection_edit
 sc_reset_connection_new
 sc_reset_global
 sc_rollback_trans
 sc_select
 sc_set_fetchmode
 sc_set_global
 sc_sql_injection
 sc_time_diff
 sc_trunc_num
 sc_url_library
 sc_user_logout
 sc_warning
 sc_webservice
 sc_where_current
 sc_where_filter
 sc_where_orig
 sc_zip_file
 Variables - Database
 Variables - Totalling

Aplicação Consulta

sc_begin_trans
 sc_calc_dv
 sc_commit_trans
 sc_concat
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_date_empty
 sc_decode
 sc_encode
 sc_exec_sql
 sc_format_num
 sc_format_num_region
 sc_lin_cod_barra_arrecadacao
 sc_lin_cod_barra_banco
 sc_lin_digitavel_arrecadacao
 sc_lin_digitavel_banco
 sc_lookup
 sc_mail_send
 sc_reset_global
 sc_rollback_trans
 sc_select
 sc_select_order
 sc_select_where
 sc_set_fetchmode
 sc_set_global
 sc_sql_injection
 sc_time_diff
 sc_trunc_num
 sc_vl_extenso
 sc_where_current
 sc_where_filter
 sc_where_orig
 Variables - Database

Aplicação ReportPDF

sc_begin_trans
 sc_calc_dv
 sc_change_connection
 sc_commit_trans
 sc_concat
 sc_connection_edit
 sc_connection_new
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_date_empty
 sc_decode
 sc_encode
 sc_exec_sql
 sc_field_disabled
 sc_format_num
 sc_get_language
 sc_get_regional
 sc_get_theme
 sc_include
 sc_lin_cod_barra_arrecadacao
 sc_lin_cod_barra_banco
 sc_lin_digitavel_arrecadacao
 sc_lin_digitavel_banco
 sc_lookup
 sc_mail_send
 sc_reset_connection_edit
 sc_reset_connection_new
 sc_reset_global
 sc_rollback_trans
 sc_select
 sc_set_fetchmode
 sc_set_global
 sc_sql_injection
 sc_time_diff
 sc_trunc_num
 sc_user_logout
 sc_warning
 sc_where_current
 sc_where_filter
 sc_where_orig
 sc_zip_file
 Variables - Database
 Variables - Totalling

onScriptInit

Aplicação Calendário

sc_apl_conf
 sc_apl_status
 sc_begin_trans
 sc_block_display
 sc_btn_copy
 sc_btn_display
 sc_calc_dv
 sc_change_connection
 sc_commit_trans
 sc_concat
 sc_connection_edit
 sc_connection_new
 sc_connection_new
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_date_empty
 sc_decode
 sc_encode
 sc_exec_sql
 sc_encode
 sc_exec_sql
 sc_exit
 sc_field_display
 sc_field_readonly
 sc_format_num
 sc_form_show
 sc_get_language
 sc_get_regional
 sc_get_theme
 sc_image
 sc_include
 sc_include_library
 sc_label
 sc_language

Aplicação Gráfico

sc_apl_conf
 sc_apl_status
 sc_begin_trans
 sc_block_display
 sc_btn_copy
 sc_btn_display
 sc_calc_dv
 sc_change_connection
 sc_commit_trans
 sc_concat
 sc_connection_edit
 sc_connection_new
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_decode
 sc_encode
 sc_exec_sql
 sc_encode
 sc_exec_sql
 sc_exit
 sc_field_display
 sc_field_readonly
 sc_format_num
 sc_get_theme
 sc_image
 sc_include
 sc_include_library
 sc_label
 sc_language
 sc_lin_cod_barra_arrecadacao
 sc_lin_cod_barra_banco
 sc_lin_digitavel_arrecadacao
 sc_lin_digitavel_banco

Aplicação Consulta

sc_apl_conf
 sc_apl_status
 sc_begin_trans
 sc_block_display
 sc_btn_copy
 sc_btn_display
 sc_calc_dv
 sc_change_connection
 sc_commit_trans
 sc_concat
 sc_connection_edit
 sc_connection_new
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_date_empty
 sc_decode
 sc_encode
 sc_exec_sql
 sc_encode
 sc_exec_sql
 sc_exit
 sc_field_display
 sc_field_readonly
 sc_format_num
 sc_get_groupby_rule
 sc_get_language
 sc_get_regional
 sc_get_theme
 sc_hide_groupby_rule
 sc_image
 sc_include
 sc_include_lib
 sc_include_library
 sc_label

Aplicação Formulário

sc_apl_conf
 sc_apl_status
 sc_begin_trans
 sc_block_display
 sc_btn_copy
 sc_btn_display
 sc_calc_dv
 sc_change_connection
 sc_commit_trans
 sc_concat
 sc_connection_edit
 sc_connection_new
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_date_empty
 sc_decode
 sc_encode
 sc_exec_sql
 sc_exit
 sc_field_display
 sc_field_readonly
 sc_format_num
 sc_format_num_region
 sc_form_show
 sc_get_language
 sc_get_regional
 sc_get_theme
 sc_image
 sc_include
 sc_include_library
 sc_label

Aplicação Controle

sc_apl_conf
 sc_apl_status
 sc_begin_trans
 sc_block_display
 sc_btn_copy
 sc_btn_display
 sc_calc_dv
 sc_change_connection
 sc_commit_trans
 sc_concat
 sc_connection_edit
 sc_connection_new
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_decode
 sc_encode
 sc_error_continue
 sc_error_exit
 sc_error_message
 sc_exec_sql
 sc_exit
 sc_field_display
 sc_field_readonly
 sc_format_num
 sc_format_num_region
 sc_form_show
 sc_get_language
 sc_get_regional
 sc_get_theme
 sc_image
 sc_include
 sc_include_library

Aplicação ReportPDF

sc_apl_conf
 sc_apl_status
 sc_begin_trans
 sc_calc_dv
 sc_change_connection
 sc_concat
 sc_connection_edit
 sc_connection_new
 sc_date
 sc_date_conv
 sc_date_dif
 sc_date_dif_2
 sc_decode
 sc_encode
 sc_exec_sql
 sc_field_color
 sc_field_display
 sc_format_num
 sc_get_theme
 sc_image
 sc_include
 sc_label
 sc_language
 sc_lin_cod_barra_arrecadacao
 sc_lin_cod_barra_banco
 sc_lin_digitavel_arrecadacao
 sc_lin_digitavel_banco
 sc_lookup
 sc_mail_send
 sc_reset_apl_conf
 sc_reset_apl_status
 sc_reset_change_connection

sc_lookup	sc_lookup	sc_language	sc_language	sc_label	sc_reset_connection_edit
sc_mail_send	sc_mail_send	sc_lin_cod_barra_arrecadacao	sc_lookup	sc_language	sc_reset_connection_new
sc_master_value	sc_make_link	sc_lin_cod_barra_banco	sc_mail_send	sc_lookup	sc_set_fetchmode
sc_redir	sc_redir	sc_lin_digitavel_arrecadacao	sc_master_value	sc_mail_send	sc_set_global
sc_reset_apl_conf	sc_reset_apl_conf	sc_lin_digitavel_banco	sc_redir	sc_master_value	sc_set_language
sc_reset_apl_status	sc_reset_apl_status	sc_log_add	sc_reset_apl_conf	sc_redir	sc_set_regional
sc_reset_change_connection	sc_reset_change_connection	sc_log_split	sc_reset_apl_status	sc_reset_apl_conf	sc_set_theme
Aplicação Calendário	Aplicação Gráfico	Aplicação Consulta	Aplicação Formulário	Aplicação Controle	Aplicação ReportPDF
sc_reset_connection_edit	sc_reset_connection_edit	sc_lookup	sc_reset_change_connection	sc_reset_apl_status	sc_site_ssl
sc_reset_connection_new	sc_reset_connection_new	sc_mail_send	sc_reset_connection_edit	sc_reset_change_connection	sc_sql_injection
sc_reset_global	sc_reset_global	sc_make_link	sc_reset_connection_new	sc_reset_connection_edit	sc_time_diff
sc_rollback_trans	sc_rollback_trans	sc_redir	sc_reset_global	sc_reset_connection_new	sc_trunc_num
sc_select	sc_select	sc_reset_apl_conf	sc_rollback_trans	sc_reset_global	sc_user_logout
sc_set_fetchmode	sc_select_field	sc_reset_apl_status	sc_select	sc_select	sc_warning
sc_set_global	sc_select_order	sc_reset_change_connection	sc_set_fetchmode	sc_set_fetchmode	sc_where_current
sc_set_language	sc_select_where	sc_reset_connection_edit	sc_set_global	sc_set_focus	sc_where_filter
sc_set_regional	sc_set_fetchmode	sc_reset_connection_new	sc_set_language	sc_set_global	sc_where_orig
sc_set_theme	sc_set_global	sc_reset_global	sc_set_regional	sc_set_language	sc_zip_file
sc_site_ssl	sc_set_language	sc_rollback_trans	sc_set_theme	sc_set_regional	Variables - Database
sc_sql_injection	sc_set_regional	sc_select	sc_site_ssl	sc_set_theme	
sc_time_diff	sc_set_theme	sc_select_field	sc_sql_injection	sc_site_ssl	
sc_trunc_num	sc_site_ssl	sc_select_order	sc_time_diff	sc_sql_injection	
sc_url_exit	sc_sql_injection	sc_select_where	sc_trunc_num	sc_time_diff	
sc_url_library	sc_time_diff	sc_set_fetchmode	sc_url_exit	sc_trunc_num	
sc_user_logout	sc_trunc_num	sc_set_global	sc_url_library	sc_url_exit	
sc_warning	sc_url_exit	sc_set_language	sc_user_logout	sc_user_library	
sc_webservice	sc_url_library	sc_set_regional	sc_warning	sc_user_logout	
sc_zip_file	sc_user_logout	sc_set_theme	sc_webservice	sc_warning	
	sc_warning	sc_site_ssl	sc_zip_file	sc_webservice	
	sc_webservice	sc_sql_injection	Variables - Authentication	sc_zip_file	
	sc_where_current	sc_sql_protect	Variables - Database		
	sc_where_filter	sc_time_diff			
	sc_where_orig	sc_trunc_num			
	sc_zip_file	sc_url_exit			
	Variables - Database	sc_url_library			
		sc_user_logout			
		sc_warning			
		sc_webservice			
		sc_where_current			
		sc_where_filter			
		sc_where_orig			
		sc_zip_file			
		Variables - Authentication			
		Variables - Database			

onLoad

Aplicação Calendário	Aplicação Formulário	Aplicação Controle	Aplicação Menu	Aplicação Menu árvore
sc_begin_trans	sc_begin_trans	sc_alert	sc_apl_conf	sc_apl_conf
sc_block_display	sc_block_display	sc_begin_trans	sc_apl_status	sc_apl_status
sc_btn_copy	sc_btn_copy	sc_block_display	sc_appmenu_add_item	sc_appmenu_add_item
sc_btn_display	sc_btn_display	sc_btn_display	sc_appmenu_create	sc_appmenu_create
sc_btn_new	sc_btn_new	sc_calc_dv	sc_appmenu_exist_item	sc_appmenu_exist_item
sc_calc_dv	sc_calc_dv	sc_change_connection	sc_appmenu_remove_item	sc_appmenu_remove_item
sc_change_connection	sc_change_connection	sc_commit_trans	sc_appmenu_reset	sc_appmenu_reset
sc_commit_trans	sc_commit_trans	sc_concat	sc_appmenu_update_item	sc_appmenu_update_item
sc_concat	sc_concat	sc_connection_edit	sc_begin_trans	sc_begin_trans
sc_connection_edit	sc_connection_edit	sc_connection_new	sc_btn_disable	sc_btn_disable
sc_connection_new	sc_connection_new	sc_date	sc_calc_dv	sc_calc_dv
sc_date	sc_date	sc_date_conv	sc_commit_trans	sc_commit_trans
sc_date_conv	sc_date_conv	sc_date_dif	sc_concat	sc_concat
sc_date_dif	sc_date_dif	sc_date_dif_2	sc_connection_edit	sc_connection_edit
sc_date_dif_2	sc_date_dif_2	sc_date_empty	sc_connection_new	sc_connection_new
sc_date_empty	sc_date_empty	sc_decode	sc_date	sc_date
sc_decode	sc_decode	sc_encode	sc_date_dif	sc_date_dif
sc_encode	sc_encode	sc_error_exit	sc_date_dif_2	sc_date_dif_2
sc_exec_sql	sc_exec_sql	sc_error_message	sc_decode	sc_decode
sc_field_disabled	sc_field_disabled	sc_exec_sql	sc_encode	sc_encode
sc_field_display	sc_field_display	sc_field_disabled	sc_exec_sql	sc_exec_sql
sc_field_readonly	sc_field_readonly	sc_field_display	sc_format_num	sc_format_num
sc_format_num	sc_format_num	sc_field_readonly	sc_format_num_region	sc_format_num_region
sc_form_show	sc_format_num_region	sc_format_num	sc_get_language	sc_get_language
sc_get_language	sc_form_show	sc_format_num_region	sc_get_regional	sc_get_regional
sc_get_regional	sc_get_language	sc_form_show	sc_get_theme	sc_get_theme
sc_get_theme	sc_get_regional	sc_get_language	sc_image	sc_image
sc_image	sc_get_theme	sc_get_regional	sc_include	sc_include
sc_include	sc_image	sc_get_theme	sc_include_library	sc_include_library
sc_include_library	sc_include	sc_image	sc_language	sc_language
sc_label	sc_include_library	sc_include_library	sc_lookup	sc_lookup
sc_lin_cod_barra_arrecadacao	sc_label	sc_label	sc_mail_send	sc_mail_send
sc_lin_cod_barra_banco	sc_lin_cod_barra_arrecadacao	sc_lookup	sc_menu_delete	sc_menu_delete
sc_lin_digitavel_arrecadacao	sc_lin_cod_barra_banco	sc_mail_send	sc_menu_disable	sc_menu_disable
sc_lin_digitavel_banco	sc_lin_digitavel_arrecadacao	sc_master_value	sc_menu_force_mobile	sc_menu_force_mobile
sc_log_split	sc_lin_digitavel_banco	sc_redir	sc_redir	sc_redir
sc_lookup	sc_log_split	sc_reset_connection_edit	sc_reset_apl_conf	sc_reset_apl_conf
sc_mail_send	sc_lookup	sc_reset_connection_new	sc_reset_apl_status	sc_reset_apl_status
sc_master_value	sc_mail_send	sc_reset_global	sc_reset_change_connection	sc_reset_change_connection
sc_redir	sc_master_value	sc_rollback_trans	sc_reset_connection_edit	sc_reset_connection_edit
sc_reset_connection_edit	sc_redir	sc_select	sc_reset_connection_new	sc_reset_connection_new
sc_reset_connection_new	sc_reset_connection_edit	sc_set_fetchmode	sc_reset_global	sc_reset_global
sc_reset_global	sc_reset_connection_new	sc_set_focus	sc_reset_menu_delete	sc_reset_menu_delete
sc_rollback_trans	sc_reset_global	sc_set_global	sc_reset_menu_disable	sc_reset_menu_disable
sc_select	sc_rollback_trans	sc_site_ssl	sc_rollback_trans	sc_rollback_trans
sc_set_fetchmode	sc_select	sc_sql_injection	sc_select	sc_select
sc_set_focus	sc_set_fetchmode	sc_sql_protect	sc_set_fetchmode	sc_set_fetchmode
sc_set_global	sc_set_focus	sc_time_diff	sc_set_global	sc_set_global
sc_site_ssl	sc_set_global	sc_trunc_num	sc_set_language	sc_set_language
sc_sql_injection	sc_site_ssl	sc_url_exit	sc_set_regional	sc_set_regional
sc_time_diff	sc_sql_injection	sc_url_library	sc_set_theme	sc_set_theme
sc_trunc_num	sc_time_diff	sc_user_logout	sc_site_ssl	sc_site_ssl
sc_url_exit	sc_trunc_num	sc_warning	sc_sql_injection	sc_sql_injection
sc_url_library	sc_url_exit	sc_webservice	sc_time_diff	sc_time_diff
sc_user_logout	sc_url_library	sc_zip_file	sc_trunc_num	sc_trunc_num
sc_warning	sc_user_logout	Variables - Database	sc_url_library	sc_url_library
sc_webservice	sc_warning		sc_warning	sc_warning
sc_zip_file	sc_webservice		sc_webservice	sc_webservice
	sc_zip_file		sc_zip_file	sc_zip_file
			Variables - Database	Variables - Database

onNavigate

Aplicação Calendário Aplicação Consulta	Aplicação Formulário	Aplicação Controle	Aplicação Formulário	Aplicação Menu	Aplicação Menu árvore
sc_begin_trans sc_calc_dv sc_commit_trans sc_concat sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_exec_sql sc_format_num sc_format_num_region sc_get_groupby_rule sc_get_language sc_get_regional sc_get_theme sc_include sc_label sc_lin_cod_barra_arrecadacao sc_lin_cod_barra_banco sc_lin_digitavel_arrecadacao sc_lin_digitavel_banco sc_lookup sc_make_link sc_rollback_trans sc_select sc_set_fetchmode sc_sql_injection sc_sql_protect sc_time_diff sc_trunc_num sc_vl_extenso sc_where_current sc_where_filter sc_where_orig Variables - Database			sc_begin_trans sc_block_display sc_btn_display sc_calc_dv sc_commit_trans sc_concat sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_exit sc_error_message sc_exec_sql sc_field_disabled sc_field_display sc_field_readonly sc_format_num sc_format_num_region sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_label sc_lin_cod_barra_arrecadacao sc_lin_cod_barra_banco sc_lin_digitavel_arrecadacao sc_lin_digitavel_banco sc_lookup sc_mail_send sc_master_value sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_site_ssl sc_sql_injection sc_time_diff sc_trunc_num sc_url_exit sc_url_library sc_user_logout sc_warning sc_webservice sc_zip_file		

onRecord

Aplicação Consulta	Aplicação Formulário	Aplicação ReportPDF
sc_begin_trans sc_calc_dv sc_commit_trans sc_concat sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_exec_sql sc_field_color sc_field_style sc_format_num sc_format_num_region sc_get_groupby_rule sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_link sc_lin_cod_barra_arrecadacao sc_lin_cod_barra_banco sc_lin_digitavel_arrecadacao sc_lin_digitavel_banco sc_lookup sc_mail_send sc_make_link sc_reset_global sc_rollback_trans sc_select sc_seq_register sc_set_fetchmode sc_set_global sc_sql_injection sc_sql_protect sc_text_style sc_time_diff sc_trunc_num sc_vl_extenso	sc_begin_trans sc_block_display sc_btn_display sc_calc_dv sc_change_connection sc_commit_trans sc_concat sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_exec_sql sc_field_disabled_record sc_field_display sc_field_readonly sc_format_num sc_format_num_region sc_get_language sc_get_regional sc_get_theme sc_image sc_include sc_include_library sc_label sc_lin_cod_barra_arrecadacao sc_lin_cod_barra_banco sc_lin_digitavel_arrecadacao sc_lin_digitavel_banco sc_lookup sc_mail_send sc_master_value sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_set_global	sc_begin_trans sc_calc_dv sc_change_connection sc_commit_trans sc_concat sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_exec_sql sc_field_color sc_get_language sc_get_regional sc_get_theme sc_include sc_link sc_lin_cod_barra_arrecadacao sc_lin_cod_barra_banco sc_lin_digitavel_arrecadacao sc_lin_digitavel_banco sc_lookup sc_mail_send sc_reset_connection_edit sc_reset_connection_new sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_set_global sc_sql_injection sc_sql_protect sc_time_diff sc_trunc_num sc_url_exit sc_user_logout sc_warning sc_where_current

sc_where_current sc_where_filter sc_where_orig Variables - Database Variables - Totalling	sc_set_global sc_site_ssl sc_sql_injection sc_time_diff sc_trunc_num sc_url_exit	sc_where_filter sc_where_orig sc_zip_file Variables - Database Variables - Totalling
Aplicação Consulta	Aplicação Formulário sc_user_logout sc_warning sc_webservice sc_zip_file	Aplicação ReportPDF

onRefresh

Aplicação Calendário	Aplicação Formulário	Aplicação Controle
sc_begin_trans sc_block_display sc_btn_display sc_calc_dv sc_commit_trans sc_concat sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_exec_sql sc_field_display sc_field_readonly sc_format_num sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_label sc_lin_cod_barra_arrecadacao sc_lin_cod_barra_banco sc_lin_digitavel_arrecadacao sc_lin_digitavel_banco sc_lookup sc_mail_send sc_master_value sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_set_focus sc_set_global sc_site_ssl sc_sql_injection sc_time_diff sc_trunc_num sc_url_exit sc_url_library sc_warning sc_webservice sc_zip_file	sc_begin_trans sc_block_display sc_btn_display sc_calc_dv sc_commit_trans sc_concat sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_exec_sql sc_field_display sc_field_readonly sc_format_num sc_format_num_region sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_label sc_lin_cod_barra_arrecadacao sc_lin_cod_barra_banco sc_lin_digitavel_arrecadacao sc_lin_digitavel_banco sc_lookup sc_mail_send sc_master_value sc_reset_global sc_rollback_trans sc_select sc_set_fetchmode sc_set_focus sc_set_global sc_site_ssl sc_sql_injection sc_time_diff sc_trunc_num sc_url_exit sc_url_library sc_warning sc_webservice sc_zip_file	sc_begin_trans sc_block_display sc_btn_display sc_calc_dv sc_changed sc_commit_trans sc_concat sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_exit sc_error_message sc_exec_sql sc_field_display sc_field_readonly sc_format_num sc_format_num_region sc_get_language sc_get_regional sc_get_theme sc_include sc_include_library sc_lookup sc_mail_send sc_master_value sc_reidir sc_reset_connection_edit sc_reset_connection_new sc_rollback_trans sc_select sc_set_fetchmode sc_set_focus sc_set_global sc_sql_injection sc_time_diff sc_trunc_num sc_url_library sc_warning sc_webservice sc_zip_file Variables - Database

onValidate

Aplicação Calendário	Aplicação Formulário	Aplicação Controle
sc_ajax_javascript sc_begin_trans sc_block_display sc_btn_delete sc_btn_display sc_btn_insert sc_btn_update sc_calc_dv sc_changed sc_commit_trans sc_concat sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_exit sc_error_message sc_exec_sql sc_field_display sc_field_readonly sc_format_num sc_form_show sc_get_language sc_get_regional sc_get_theme sc_image sc_include sc_include_library sc_ldap_login sc_ldap_logout sc_ldap_search sc_log_add	sc_ajax_javascript sc_begin_trans sc_block_display sc_btn_delete sc_btn_display sc_btn_insert sc_btn_update sc_calc_dv sc_changed sc_commit_trans sc_concat sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_date_empty sc_decode sc_encode sc_error_exit sc_error_message sc_exec_sql sc_field_display sc_field_readonly sc_format_num sc_format_num_region sc_form_show sc_get_language sc_get_regional sc_get_theme sc_image sc_include sc_include_library sc_ldap_login sc_ldap_logout sc_ldap_search	sc_alert sc_apl_conf sc_apl_status sc_begin_trans sc_btn_display sc_calc_dv sc_commit_trans sc_concat sc_connection_edit sc_connection_new sc_date sc_date_conv sc_date_dif sc_date_dif_2 sc_decode sc_encode sc_error_exit sc_error_message sc_exec_sql sc_field_display sc_field_readonly sc_format_num_region sc_get_theme sc_include sc_include_library sc_language sc_ldap_login sc_ldap_logout sc_ldap_search sc_lin_cod_barra_arrecadacao sc_lin_cod_barra_banco sc_lin_digitavel_arrecadacao sc_lin_digitavel_banco sc_lookup sc_mail_send

sc_lookup	sc_log_add	sc_make_link
sc_mail_send	sc_lookup	sc_redir
sc_master_send	sc_mail_send	sc_reset_apl_conf
sc_redir	sc_master_value	sc_reset_apl_status
sc_reset_global	sc_redir	sc_reset_change_connection
sc_rollback_trans	sc_reset_global	sc_reset_connection_edit
sc_select	sc_rollback_trans	sc_reset_connection_new
Aplicação Calendário	Aplicação Formulário	Aplicação Controle
sc_set_fetchmode	sc_select	sc_reset_global
sc_set_global	sc_set_fetchmode	sc_rollback_trans
sc_site_ssl	sc_set_global	sc_select
sc_sql_injection	sc_site_ssl	sc_set_fetchmode
sc_sql_protect	sc_sql_injection	sc_set_global
sc_time_diff	sc_sql_protect	sc_set_language
sc_trunc_num	sc_time_diff	sc_set_regional
sc_url_exit	sc_trunc_num	sc_set_theme
sc_url_library	sc_url_exit	sc_site_ssl
sc_user_logout	sc_url_library	sc_sql_injection
sc_warning	sc_user_logout	sc_sql_protect
sc_webservice	sc_warning	sc_time_diff
sc_zip_file	sc_webservice	sc_trunc_num
	sc_zip_file	sc_url_exit
		sc_url_library
		sc_user_logout
		sc_warning
		sc_webservice
		sc_zip_file
		Variables - Database

onValidateFailure

Aplicação Calendário	Aplicação Formulário	Aplicação Controle
sc_ajax_javascript	sc_ajax_javascript	sc_ajax_javascript
sc_begin_trans	sc_begin_trans	sc_apl_conf
sc_block_display	sc_block_display	sc_apl_status
sc_btn_delete	sc_btn_delete	sc_begin_trans
sc_btn_display	sc_btn_display	sc_btn_display
sc_btn_insert	sc_btn_insert	sc_calc_dv
sc_btn_update	sc_btn_update	sc_commit_trans
sc_calc_dv	sc_calc_dv	sc_concat
sc_changed	sc_changed	sc_connection_edit
sc_commit_trans	sc_commit_trans	sc_connection_new
sc_date	sc_date	sc_date
sc_date_conv	sc_date_conv	sc_date_conv
sc_date_dif	sc_date_dif	sc_date_dif
sc_date_dif_2	sc_date_dif_2	sc_date_dif_2
sc_date_empty	sc_date_empty	sc_decode
sc_decode	sc_decode	sc_encode
sc_encode	sc_encode	sc_error_exit
sc_error_exit	sc_error_exit	sc_error_message
sc_error_message	sc_error_message	sc_exec_sql
sc_exec_sql	sc_exec_sql	sc_field_display
sc_field_display	sc_field_display	sc_field_readonly
sc_field_readonly	sc_field_readonly	sc_format_num
sc_format_num	sc_format_num	sc_format_num_region
sc_get_language	sc_format_num_region	sc_get_language
sc_get_regional	sc_get_language	sc_get_regional
sc_get_theme	sc_get_regional	sc_get_theme
sc_include	sc_get_theme	sc_image
sc_include_library	sc_include	sc_include
sc_mail_send	sc_include_library	sc_include_library
sc_master_value	sc_mail_send	sc_lookup
sc_reset_global	sc_master_value	sc_mail_send
sc_rollback_trans	sc_reset_global	sc_redir
sc_select	sc_rollback_trans	sc_reset_connection_edit
sc_set_fetchmode	sc_select	sc_reset_connection_new
sc_set_global	sc_set_fetchmode	sc_reset_global
sc_site_ssl	sc_set_global	sc_rollback_trans
sc_sql_injection	sc_site_ssl	sc_select
sc_time_diff	sc_sql_injection	sc_set_fetchmode
sc_trunc_num	sc_time_diff	sc_set_global
sc_url_exit	sc_trunc_num	sc_site_ssl
sc_url_library	sc_url_exit	sc_sql_injection
sc_warning	sc_url_library	sc_time_diff
sc_webservice	sc_warning	sc_trunc_num
sc_zip_file	sc_webservice	sc_url_exit
	sc_zip_file	sc_url_library
		sc_warning
		sc_webservice
		sc_zip_file
		Variables - Database

onValidateSuccess

Aplicação Calendário	Aplicação Formulário	Aplicação Controle
sc_ajax_javascript	sc_ajax_javascript	sc_apl_conf
sc_begin_trans	sc_begin_trans	sc_apl_status
sc_block_display	sc_block_display	sc_begin_trans
sc_btn_delete	sc_btn_delete	sc_calc_dv
sc_btn_display	sc_btn_display	sc_commit_trans
sc_btn_insert	sc_btn_insert	sc_concat
sc_btn_update	sc_btn_update	sc_connection_edit
sc_calc_dv	sc_calc_dv	sc_connection_new
sc_changed	sc_changed	sc_date
sc_commit_trans	sc_commit_trans	sc_date_conv
sc_concat	sc_concat	sc_date_dif
sc_connection_edit	sc_connection_edit	sc_date_dif_2
sc_connection_new	sc_connection_new	sc_decode
sc_date	sc_date	sc_encode
sc_date_conv	sc_date_conv	sc_error_exit
sc_date_dif	sc_date_dif	sc_error_message
sc_date_dif_2	sc_date_dif_2	sc_exec_sql

sc_date_empty	sc_date_empty	sc_format_num
sc_decode	sc_decode	sc_format_num_region
sc_encode	sc_encode	sc_get_language
sc_error_exit	sc_error_exit	sc_get_regional
sc_error_message	sc_error_message	sc_get_theme
sc_exec_sql	sc_exec_sql	sc_image
sc_field_display	sc_field_display	sc_include
Aplicação Calendário	Aplicação Formulário	Aplicação Controle
sc_field_readonly	sc_field_readonly	sc_include_library
sc_format_num	sc_format_num	sc_language
sc_get_language	sc_format_num_region	sc_lin_cod_barra_arrecadacao
sc_get_regional	sc_get_language	sc_lin_cod_barra_banco
sc_get_theme	sc_get_regional	sc_lin_digitavel_arrecadacao
sc_include	sc_get_theme	sc_lin_digitavel_banco
sc_include_library	sc_include	sc_lookup
sc_lin_cod_barra_arrecadacao	sc_include_library	sc_mail_send
sc_lin_cod_barra_banco	sc_lin_cod_barra_arrecadacao	sc_make_link
sc_lin_digitavel_arrecadacao	sc_lin_cod_barra_banco	sc_redir
sc_lin_digitavel_banco	sc_lin_digitavel_arrecadacao	sc_reset_apl_conf
sc_lookup	sc_lin_digitavel_banco	sc_reset_apl_status
sc_mail_send	sc_lookup	sc_reset_change_connection
sc_redir	sc_mail_send	sc_reset_connection_edit
sc_reset_connection_edit	sc_redir	sc_reset_connection_new
sc_reset_connection_new	sc_reset_connection_edit	sc_reset_global
sc_reset_global	sc_reset_connection_new	sc_rollback_trans
sc_rollback_trans	sc_reset_global	sc_select
sc_select	sc_rollback_trans	sc_set_fetchmode
sc_set_fetchmode	sc_select	sc_set_global
sc_set_focus	sc_set_fetchmode	sc_set_language
sc_set_global	sc_set_focus	sc_set_regional
sc_site_ssl	sc_set_global	sc_set_theme
sc_sql_injection	sc_site_ssl	sc_site_ssl
sc_time_diff	sc_sql_injection	sc_sql_injection
sc_trunc_num	sc_time_diff	sc_time_diff
sc_url_exit	sc_trunc_num	sc_trunc_num
sc_url_library	sc_url_exit	sc_url_exit
sc_user_logout	sc_url_library	sc_url_library
sc_warning	sc_user_logout	sc_user_logout
sc_webservice	sc_warning	sc_warning
sc_zip_file	sc_webservice	sc_webservice
	sc_zip_file	sc_zip_file
		Variables - Database