****

**QtRptDesigner**

Version 3.0.0

Programmer*: Oleksii Osypov (Legacy transcription - Aleksey Osipov)*

Email*:* [*aliks-os@ukr.net*](mailto:aliks-os@ukr.net)

*Web-site:* [*http://www.qtrpt.tk*](http://www.qtrpt.tk)

*Address in Facebook* [*https://www.facebook.com/qtrpt*](https://www.facebook.com/qtrpt)

Contents

[Introduction 3](#_Toc135490829)

[Interface of the Designer 3](#_Toc135490830)

[Control keys 4](#_Toc135490831)

[Mouse control 4](#_Toc135490832)

[Edit Panel 4](#_Toc135490833)

[Formatting Panel 5](#_Toc135490834)

[Items Panel 6](#_Toc135490835)

[Settings 6](#_Toc135490836)

[Bands 7](#_Toc135490837)

[Report structure and field’s property 10](#_Toc135490838)

[Page settings 10](#_Toc135490839)

[Working with the Fields 11](#_Toc135490840)

[Text formatting 12](#_Toc135490841)

[HTML Rendering 14](#_Toc135490842)

[Field’s property and possible values 14](#_Toc135490843)

[Band’s property and possible values 16](#_Toc135490844)

[Data group property 17](#_Toc135490845)

[Barcode property 18](#_Toc135490846)

[Field highlighting 19](#_Toc135490847)

[Rich Text field editor 20](#_Toc135490848)

[Charts 20](#_Toc135490849)

[Script editor 21](#_Toc135490850)

[Plugins 25](#_Toc135490851)

[How to create custom plugin 25](#_Toc135490852)

[Plugins. Notes for AppMill users 26](#_Toc135490853)

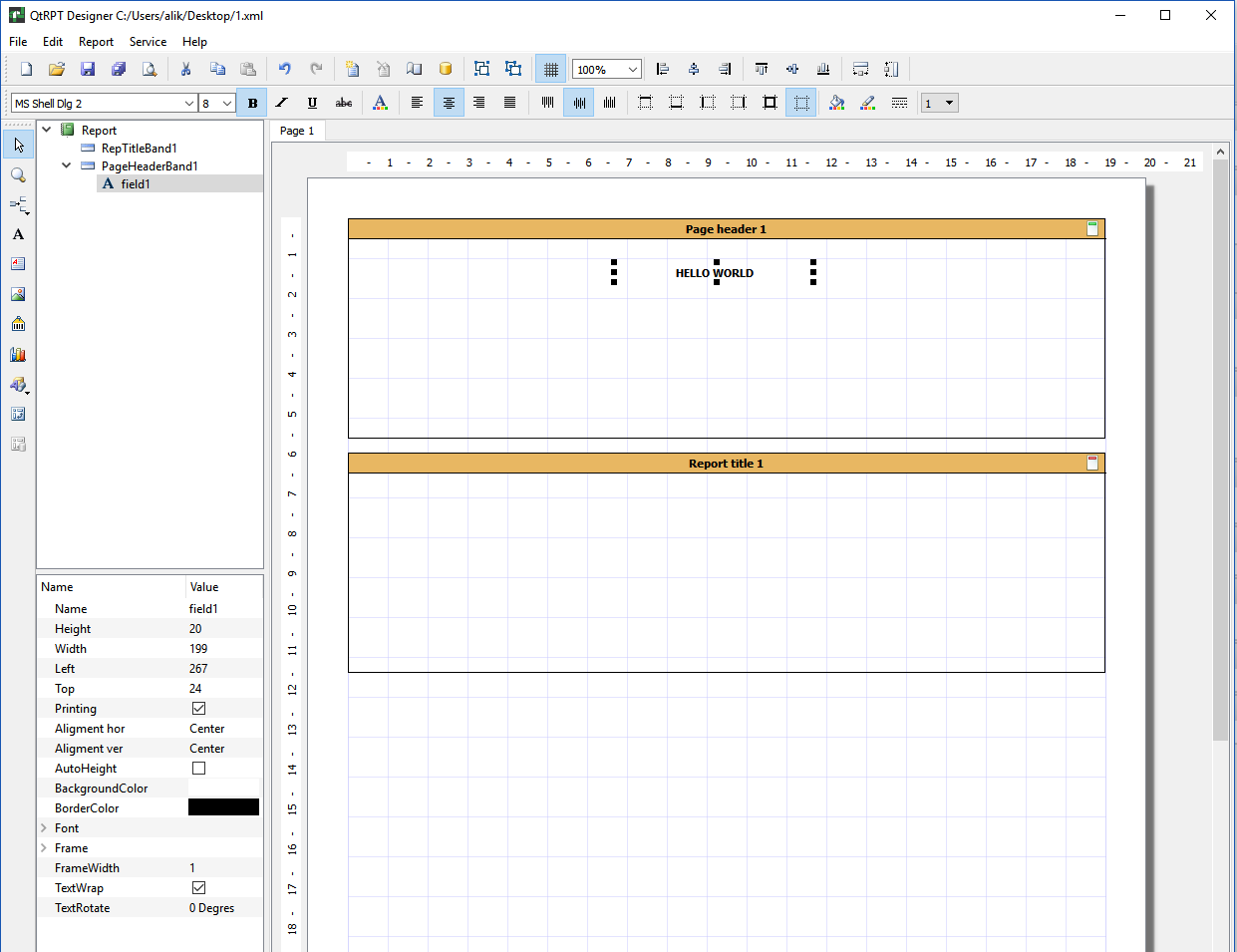
[Access to the plugins 27](#_Toc135490854)

[Edit input parameters plugin. 27](#_Toc135490855)

[Templates Directory plugin. 28](#_Toc135490856)

# Introduction

QtRptDesigner attended for preparation XML files, which will be used and processed by QtRPT engine for building reports. XML file contains information about of different items and their properties, and how this items should get data from data source, etc. QtRPT engine allows to process script embedded into the report. By script, the user can defined, how the fields must be processed depends of some condition.



# Interface of the Designer

For convenience, the buttons are grouped into a logical group and each is located on a separate panel. In addition, in the designer all elements are presented in the form of a tree, which allows you to clearly see the hierarchy of the report.

# Control keys

Different key’s combination, allows easy navigation and control of field’s size.

|  |  |
| --- | --- |
| Keys | Description |
| Ctrl + O | File -> Open menu command |
| Ctrl + S | File -> Save menu command |
| Ctrl + Z | Undo command |
| Ctrl + Y | Redo command |
| Ctrl + C | Copy command |
| Ctrl + X | Cut command |
| Ctrl + V | Paste command |
| Ctrl + arrows | Move selected object(s) |
| Shift + arrows | Modify sizes of selected object(s) |
| Del | Delete selected object(s) |

# Mouse control

|  |  |
| --- | --- |
| Operation | Description |
| Ctrl + left button | Select by clicking one field. Hold pressed Ctrl, click on another field(s) to select a several fields. |
| Double click | Open editor for the object |
| Left button | Select the object |

## Edit Panel



Contains the following buttons

|  |  |
| --- | --- |
|  | New report |
|  | Open |
|  | Save |
|  | Save as |
|  | Preview |
|  | Cut |
|  | Copy |
|  | Paste |
|  | Undo |
|  | Redo |
|  | New report page |
|  | Delete page of the report |
|  | Page setting |
| D:\Projects\QtRptProject\QtRptDesigner\images\script.png | Script editor |
|  | Data source ***(Not needed for Codiac users)*** |
|  | Group selected items |
|  | Ungroup |
|  | Show grid |
|  | Align by left edge all selected items |
|  | Align by center all selected items |
|  | Align by right edge all selected items |
|  | Align by top edge all selected items |
|  | Align by center (vertical) edge all selected items |
|  | Align by bottom edge all selected items |
|  | Set same width on selected items |
|  | Set same height on selected items |

## Formatting Panel

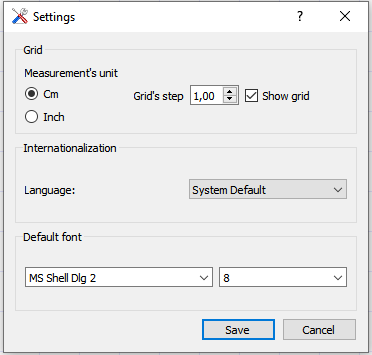


|  |  |
| --- | --- |
|  | Font selection |
|  | Bold |
|  | Italic |
|  | Underline |
|  | Strikeout |
|  | Font color |
|  | Align left |
|  | Align center |
|  | Align right |
|  | Justify |
|  | Align top |
|  | Align V center |
|  | Align bottom |
|  | Top border |
|  | Bottom border |
|  | Left border |
|  | Right border |
|  | Set all borders |
|  | Remove all borders |
|  | Background color |
|  | Border color |
|  | Border style |
|  | Border thickness |

## Items Panel

|  |  |
| --- | --- |
|  | Select tool |
|  | Magnifying glass |
|  | Add band |
|  | Add field |
|  | Add rich text |
|  | Add image |
|  | Add barcode |
|  | Add chart |
|  | Add draw |
| D:\Projects\QtRptProject\QtRptDesigner\images\crossTab.png | Add cross table ***(Not needed for Codiac users)*** |

# Settings



The “Settings” Available via “Service - > Settings” menu.

On the dialog page you can setup the Designer options such as Measurement’s unit of the grid, language of the user interface, default font for the Text fields.

## Bands

Band is area where the user’s fields can are placed. The band can be a following type:

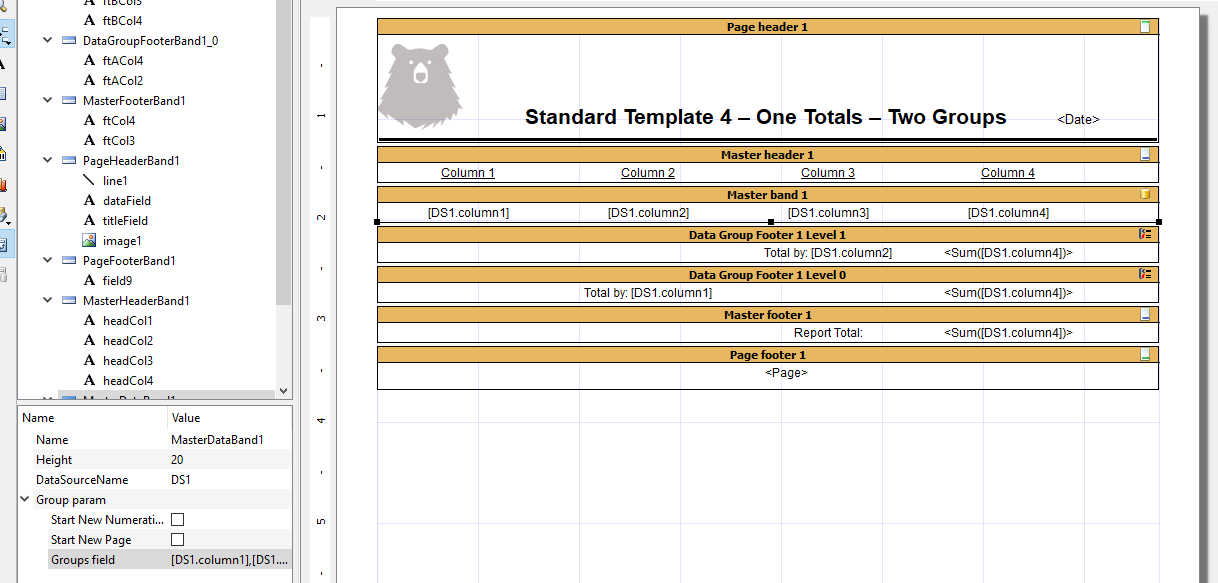
|  |  |
| --- | --- |
| **Band type** | **Purpose of the band** |
| Report Title | This band is printed only once at the start of the report and only on the first page |
| Page Header | This band is printed at the top on each page |
| Data Grouping Header | This band is used when you need to group data that are placed on the Master Data band and show Header for each group |
| Master Header | This band is printed before Master Data band |
| Master Data | This band is intended for printing of some data set. The band printing is repeated depends of records count |
| Master Footer | This band is printed after Master Data band |
| Data Grouping Footer | This band is used when you need to group data that are placed on the Master Data band and show Footer for each group |
| Page Footer | This band is printed at the bottom on each page |
| Report Summary | This band is printed only once at the end of the report and only on the last page |

To make report more flexible, you can put more than Master Data Band and corresponded bands (Master Header, Master Footer and Grouping Bands). Each Master Data Band has own data source. Each Master Data Band may have only one corresponded Data Grouping Header and from one to three corresponded Data Grouping Footer Bands which allow to show aggregate values on different level of grouping (max 3 levels).

The example below, show report which consists of

* Page Header Band – here is placed the logo and report title.
* Master Header Band – here is placed the titles of the columns
* Master Data Band - here is placed fields with alias of the data source
* Data Group Footer Band Level 1 – here are placed the fields for aggregate function. It used for calculating sub-total for values of 1 level
* Data Group Footer Band Level 0 - here are placed the fields for aggregate function. It used for calculating sub-total for values of 0 level.
* Master Footer Band – this band will be shown after the last data row. You can use it for calculating of aggregate function such as Total of all report
* Page Footer

To indicate by which aliases carry out the grouping, and order of grouping, please select Master Data Band. Then in the group expand “Group Param” section. And select “Groups field” parameter. Here, you may enter the aliases for grouping. In the example, you may see: [DS1.column1],[DS1.column2]



Lets go into deep how grouping works.

|  |  |  |  |
| --- | --- | --- | --- |
| Column1 (city) | Column2 (goods) | Column3 (qty) | Column4 (money) |
| London | Banana | 10 | 100 |
| Paris | Orange | 20 | 200 |
| NY | Banana | 30 | 300 |
| Paris | Orange | 40 | 400 |
| London | Banana | 50 | 500 |
| London | Orange | 60 | 600 |
| Paris | Banana | 70 | 700 |

Lets make grouping by city and goods. In the Groups field we indicate [Column1], [Column2]. The order of field means that Column1 is a level 0, Column 1 is a level 1.

In the table below we get the following result:

|  |  |  |  |
| --- | --- | --- | --- |
| London | Banana | 10 | 100 |
|  | Banana | 50 | 500 |
|  | Sub-total (Level 1) | 60 | 600 |
| London | Orange | 60 | 600 |
|  | Sub-total (Level 1) | 60 | 600 |
|  | Sub-total (Level 0) | 120 | 1200 |
| Paris | Banana | 70 | 700 |
|  | Sub-total (Level 1) | 70 | 700 |
| Paris | Orange | 20 | 20 |
|  | Sub-total (Level 1) | 20 | 200 |
|  | Sub-total (Level 0) | 90 | 900 |
| NY | Banana | 30 | 300 |
|  | Sub-total (Level 1) | 30 | 300 |
|  | Sub-total (Level 0) | 30 | 300 |
|  | Total by report (Master footer) | 280 | 2800 |

Otherwise, you can change order of Grouping fields [Column2], [Column1], so in this case we get the following result

|  |  |  |  |
| --- | --- | --- | --- |
| Banana | London | 60 | 600 |
|  | Sub-total (Level 1) | 60 | 600 |
|  | Paris | 70 | 700 |
|  | Sub-total (Level 1) | 70 | 300 |
|  | NY | 30 | 300 |
|  | Sub-total (Level 1) | 30 | 300 |
|  | Sub-total (Level 0) | 160 | 160 |
| Orange | London | 60 | 600 |
|  | Sub-total (Level 1) | 60 | 600 |
|  | Paris | 20 | 200 |
|  | Sub-total (Level 1) | 20 | 200 |
|  | Sub-total (Level 0) | 80 | 800 |
|  | Total by report (Master footer) | 280 | 2800 |

Please note, that the order of fields on report (bands) you must move as well.

# Report structure and field’s property

|  |  |
| --- | --- |
|  | At left from report you can see the report structure and set of property. When some field or band is selected, the appropriate properties are shown. From the list of properties, you can change any property, as well as by pressing appropriate button on the toolbar panel |

# Page settings

|  |  |
| --- | --- |
|  | The “Page settings” are available via “Report->Page Setting” main menu. You can set parameters for each page of the report.  Set a size and orientation, margins and background images (watermark). |

# Working with the Fields

The purpose of the field (element) is display information. The field should be placed on the band only. There are several types of the fields. Before placing field on the report, it should, contains at least one band. To place field on the band, click on selected field on the Panel of Elements, then click on band. The field will be placed. On selecting, the field’s properties will be shown in the list at the right. You can change the property of the field in the list or by clicking appropriate button on the panel.

**Fields moving and resizing**

The field can be moved or resized by mouse or by keyboard. When you use the keyboard, press **Ctrl + Arrow** key to move the field or **Shift + Arrow** to resize the field. You can select the several fields, to do it, please click on the first field, then press and hold on Ctrl key. While the Ctrl is pressed, click on the other fields that you want to select. Then you can use appropriate combination of keys (Ctrl + Arrow) or (Shift + Arrow) to move or resize the selected fields simultaneously.

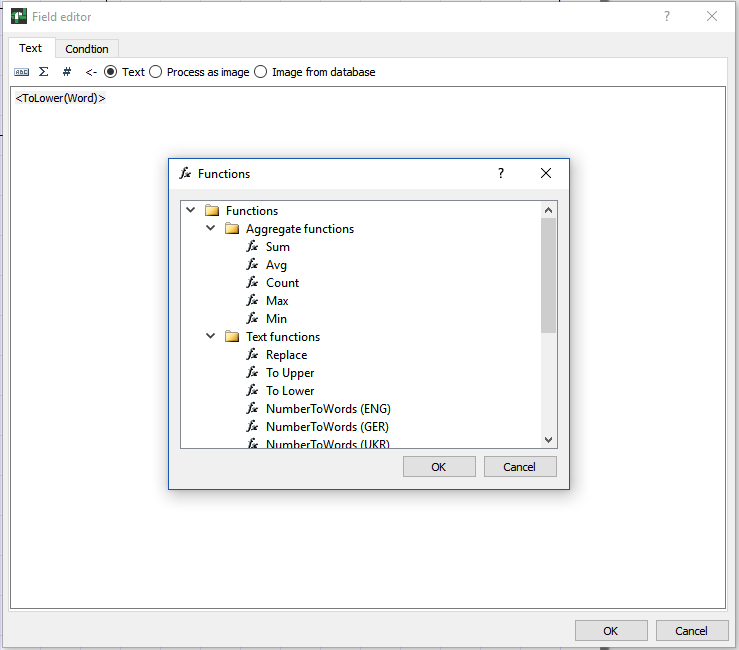
**Field’s Alignment**

Several selected fields can be aligning to the one of edge. To do it, select the several fields and click on the appropriate button. All fields will be aligned to edge of the first selected field.

**Data of the field**

If the text in the should be static – just enter the text. If the text should be taken from any data source of third party application, you should enter the alias in the following form

[AliasOfTheField] – the square brackets indicate that it is alias. In the field you can some build-in functions. To add function or take a look at the full list of function, double click on the field, then click on “Add function” button. The angle brackets indicate that it is a “function”.



You can mix using aliases and build-in functions. For example:

<ToLower([companyName])> - This means that will be taken string from alias “companyName” and all cases will be set to lower.

The price is <NumberToWords(‘ENG’, [price])> dollars - This means that will be take value from alias “price” and digital value will be written as a string. For example, if the price is 100, then the result of processing will be as “The price is one hundred dollars”.

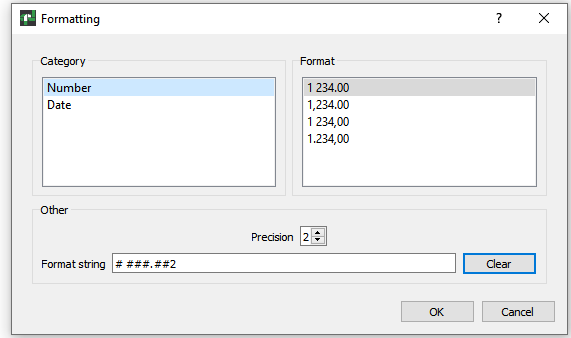
You can also use some system variables:

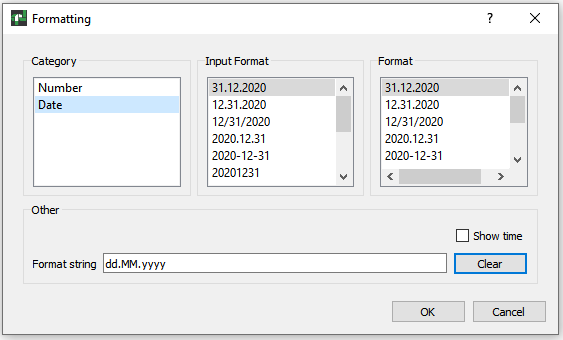
* Date
* Time
* Page
* TotalPages
* LineNo
* LineCount

During the report generations, the appropriate value will be inserted into the field. For example,

The page <Page> of <TotalPages> - Will generate: “The page 1 of 3”.

## Text formatting





It is possible to apply Number and Date formatting to the field.

The Number formatting can be for 4 cases. The format defines the sign between the decimal part, the number of decimal places, and the character between thousands.

Date formatting is a little more complicated. "Format" is the format of the date as it should be displayed. Date data can come from various sources such as Application, Database, etc. The date format in these sources may differ from the format you want to display. "Input Format" is the date format used in the data source.

## HTML Rendering

If the for the Text field object the property HTML Rendering is marked, the object does understand some simple HTML tags. Tags can belocated within the text of the object. Tags are disabled by default, but to enable them either select “HTML Renderig” in the property in the object inspector.

Here is the list of supported tags:

|  |  |
| --- | --- |
| Tag | Description |
| <b> | Bold text |
| <i> | Italic text |
| <u> | Underlined text |
| <s> | Crossed text |
| <sub> | Subscript |
| <sup> | Superscript |

Please note that only a few tags are supported, but this should be enough for the majority of applications. It is not possible to modify the font size or name by means of HTML tags. The following examples show how these tags can be used.

text in <b>bold text</b>312

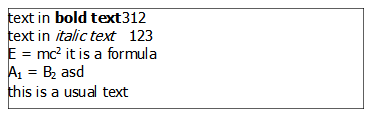
text in <i>italic text</i> 123

E = mc<sup>2</sup> it is a formula

A<sub>1</sub> = B<sub>2</sub> asd

this is a usual text

And result of rendering:



# Field’s property and possible values

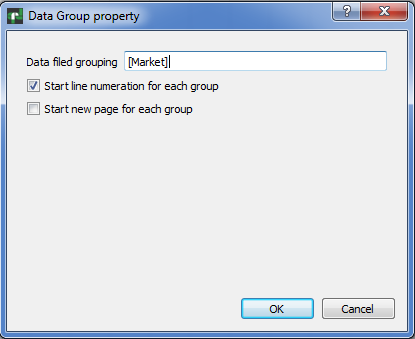
Field’s property

|  |  |  |
| --- | --- | --- |
| Field’s type | Property | Note |
| Text field | Name | Name of the object |
|  | Height | Height of the object |
|  | Width | Width of the object |
|  | Left | Left position of the object |
|  | Top | Top position of the object |
|  | Printing | Mark that object is printable |
|  | Aligment hor | Left, Right, Center, Justify |
|  | Aligment ver | Top, Center, Bottom |
|  | Auto height | The height of the object will be adjusted to the max height in the band |
|  | HTML rendering | HTML tags will be processed |
|  | Background color | Background color |
|  | Border color | Border color |
|  | Font | Font’s property |
|  | Frame | Indicate which part of border will be drawn (Top, bottom, left, right) |
|  | Frame width | Thickness of the border |
|  | Text wrap | Text wrapping |
|  | Text rotate | Angle of the text rotation. 0, 90, 180, 270 degres |
| Rich tex | Name | Name of the object |
|  | Height | Height of the object |
|  | Width | Width of the object |
|  | Left | Left position of the object |
|  | Top | Top position of the object |
|  | Printing | Mark that object is printable |
| Image | Name | Name of the object |
|  | Height | Height of the object |
|  | Width | Width of the object |
|  | Left | Left position of the object |
|  | Top | Top position of the object |
|  | Printing | Mark that object is printable |
|  | Aligment hor | Left, Right, Center, Justify |
|  | Aligment ver | Top, Center, Bottom |
|  | IgnoreRatioAspect | If marked, the aspect ratio will be ignored |
| Barcode | Name | Name of the object |
|  | Height | Height of the object |
|  | Width | Width of the object |
|  | Left | Left position of the object |
|  | Top | Top position of the object |
|  | Printing | Mark that object is printable |
|  | BarcodeType | Different types of the barcode |
|  | BarcodeFrameType | Print border around the barcode |
| Diagram | Name | Name of the object |
|  | Height | Height of the object |
|  | Width | Width of the object |
|  | Left | Left position of the object |
|  | Top | Top position of the object |
|  | Printing | Mark that object is printable |
| Line | Name | Name of the object |
|  | FrameWidth | Thickness of the line |
|  | BorderColor | Color of the line |
|  | Length | Line’s lenght |
|  | Printing | Mark that object is printable |
|  | ArrowStart | Draw arrow at the start of line |
|  | ArrowEd | Draw arrow at the end of line |
| Reactangle (Rouded).  Elipse. Triangle. Rhombus. | Name | Name of the object |
|  | Height | Height of the object |
|  | Width | Width of the object |
|  | Left | Left position of the object |
|  | Top | Top position of the object |
|  | Printing | Mark that object is printable |
|  | Background color | Background color |
|  | Border color | Border color |
|  | Frame width | Thickness of the border |

# Band’s property and possible values

|  |  |  |
| --- | --- | --- |
| Band’s type | Property | Note |
| Report title | Name | Name of the object |
|  | Height | Height of the object |
| Page header | Name | Name of the object |
|  | Height | Height of the object |
| Data Group header | Name | Name of the object |
|  | Height | Height of the object |
| Master Header | Name | Name of the object |
|  | Height | Height of the object |
|  | Show In Group | If marked, this band will placed inside of each Data group, otherwise this will be on the top of the Data Group |
| Master Data | Name | Name of the object |
|  | Height | Height of the object |
|  | DataSourceName | The name of data source from which the data will be taken |
|  | Group’s field | Field’s alias by which the grouping is carry out |
|  | Start New numeration | If marked, the line numeration will be different for each group |
|  | Start New Page | If marked, the each group will be started from new page |
| Master Footer | Name | Name of the object |
|  | Height | Height of the object |
|  | Show In Group | If marked, this band will placed inside of each Data group, otherwise this will be on the top of the Data Group |
| Data Group Footer | Name | Name of the object |
|  | Height | Height of the object |
|  | Group Level | Group nesting number. There can be a maximum of three levels of nesting |
| Page Footer | Name | Name of the object |
|  | Height | Height of the object |
| Report Summary | Name | Name of the object |
|  | Height | Height of the object |

## Data group property

****

Grouping of the data can be in the range 0-2. For each of group can be set footer, where can be set the fields for calculating aggregate values such as AVG, SUM, COUNT, MIN and MAX

## Barcode property

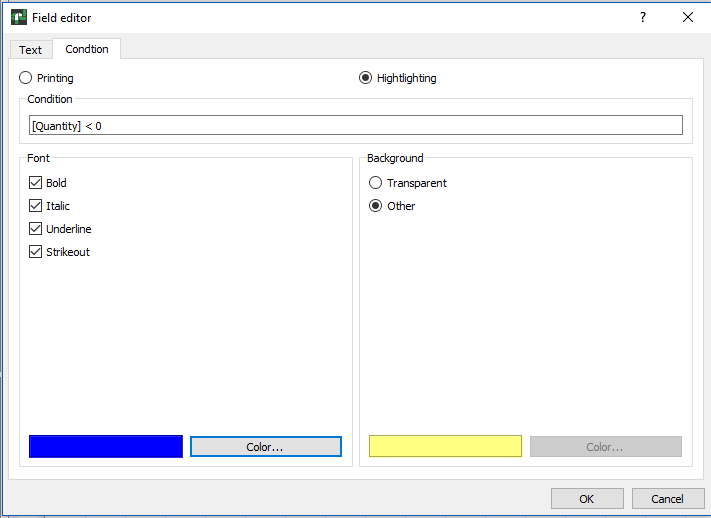
***Please note*: that since version 1.4.5, to use barcode feature, you need to have built QtZint library. The source files are in folder Zint-2.4.4. After building, place (QtZint.dll, QtZint.so file) library into the folder where your application able to find it.**

****

## Field highlighting

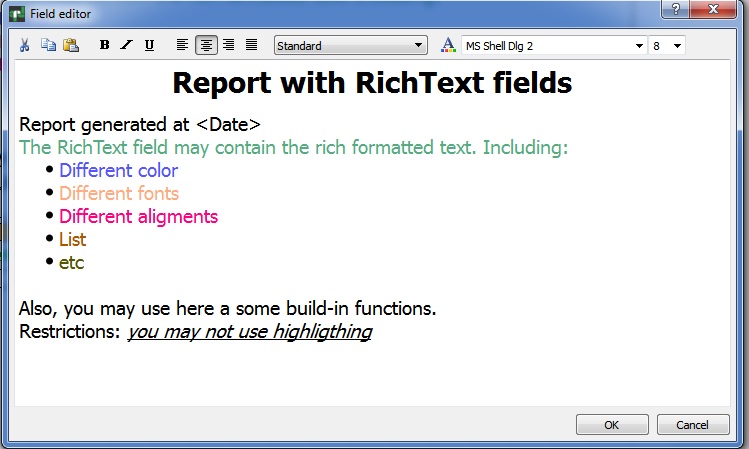
You can highlight the field depends of value. You should switch to “Condition” tab, click on “Highlight” and enter some condition, for example

[Condition] < 0, so when alias Condition will have value below zero, the font color will be blue and background color will be yellow in our example. You can set font Bold, Italic, underline or Strikeout.



# Rich Text field editor

To edit the field “Rich text”, double click on the chosen container. The dialog box in which you can bring the text will open, change its property. You may change font’s property such as Bold, Italic, Underline, change color of the text. This field also supports system varibales and work with external data. However, this field doesn’t works with conditions.

****

# Charts

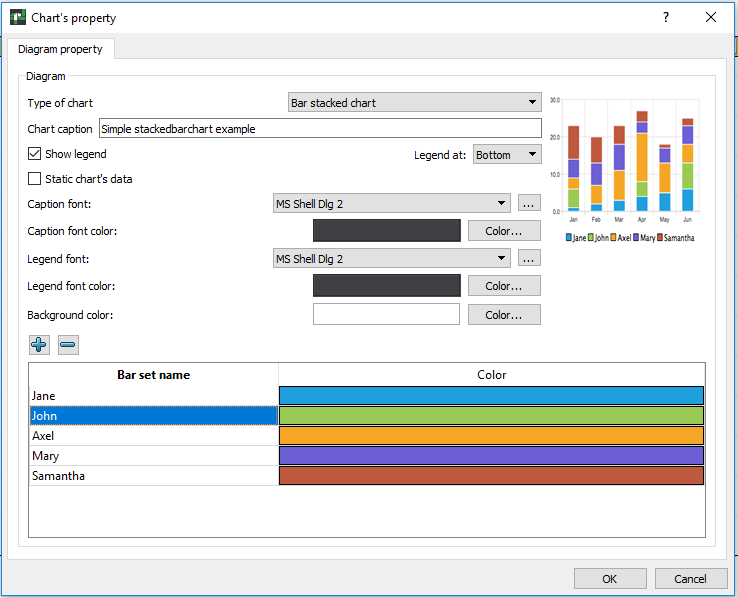
The QtRptDesigner allows to work with a different type of charts, the examples below.

|  |  |
| --- | --- |
|  |  |
|  |  |

To edit the properties of the chart, you should double click on the chart. Will be open the dialog with various fields to setup the properties of the chart. The chart can be static, i.e. value stay un-changed or dynamic, i.e. the values for diagram will get from data source.

***Note for Codiac Users***: For dynamic chart, you should enter the *alias* for data source of the chart (series).

For Stacked bar chart you should setup data source for each series. You can setup various font, font color, background color, captions, etc.



# Script editor

Script editor allows to enter scripts, which will be executed during report building. Script allows to carry out the some control of process of report building. The following objects are available in the script engine.

**QtRPT** - name of QtRPT instance in the script engine

has following properties, members and functions

- pageList - list of the RptPageObject

**RptPageObject -** represents report page, and it has the following properties, and functions

- setVisible(bool value) - makes report report page visible/invisible

**RptFieldObject**

- value - value of the filed, including variables

- width - width of the field

- height - height of the field

- top - top of the field

- left - left of the field

- visible - visibility of the field

- rotate - rotation of the field

- fontColor - color of the font

- backgroundColor - background color

The examples of some command:

To get the list of report pages

*debug(QtRPT.pageList);*

To get count of report pages

*debug(QtRPT.pageList.length);*

To get the report page name

*debug(QtRPT.pageList[0].objectName);*

**Common functions**

for debugging, you can use print or debug function

print in QtCreator word TEST + value of variable [var1]

*debug('TEST:' + [var1]);*

*print('TEST:' + [var1]);*

**Common section**

- not depends from order of the fields and evaluate as

as report starts

checking the value of variable var1 from user application

and makes report page visible/invisible depends of value.

Please note, that here only first value of variable will be taken

If you want to process each value of variable from data set, you must

place such processing in the function of appropriate field

*if ([var1] > 1000)*

*QtRPT.pageList[0].setVisible(false);*

*else*

*QtRPT.pageList[0].setVisible(true);*

To process field, you must define function with name field1 before getting data from user application

**VERY IMPORTANT!**

The name functions must be as nameOfField+BeforeData

or nameOfField+AfterData

*BeforeData functions will be executed before start of processing appropriate field*

*AfterData function will be executed after getting data from user application*

***Below some examples of using***

function field1BeforeData()

{

// making the field visible/invisible depends of value

if ([var1] > 1000) {

field1.visible = false;

field1.value = 'HELLO, I am invisible'; // assigning the value to the field

} else {

field1.visible = true;

}

}

/\*Processing field with name field1 after getting data from user application

Please note, that use After function preferably only for changing text value of the field

\*/

function field1AfterData(value)

{

/\*This function mainly intended for modifing of result string.

DONT use the following settings in AFTER section

Field1.visible = false;

If you DONT want to change value string, just comment 'return'

Here we modified the value that comes from user application and add

the string to it\*/

field1.fontColor = QColor(255,0,0);

return value + " It is corrected string";

}

/\*Here we rotate the field\*/

function field2BeforeData()

{

var w = field2.width;

var h = field2.height;

field2.rotate = 3;

field2.height = w;

field2.width = h;

}

/\*Here we change the width of field3, set the same as field4 has\*/

function field3BeforeData()

{

field3.value = 'We change width of the field';

field3.width = field4.width;

}

/\*Here we set the font and background color\*/

function field4BeforeData()

{

field4.value = 'We change the color of font and background';

field4.fontColor = QColor(0,255,0);

field4.backgroundColor = QColor(255,0,0);

}

/\*Making field invisible\*/

function field6BeforeData()

{

field6.visible = false;

}

# Plugins

The QtRptDesigner functionality can be enhanced via plugin technology. Some plugins can be automatically launched on QtRptDesigner start, some can be run via Main Menu -> Service -> Plugins -> Name of Plugin. Possible to define should plugin execute some function before report preview or not.

## How to create custom plugin

To create custom plugin, you must inherit the CustomInterface like in the code below

#pragma once

#include <QObject>

#include <QtPlugin>

#include "CustomInterface.h"

class SessionPlugin : public QObject, CustomInterface

{

Q\_OBJECT

Q\_PLUGIN\_METADATA(IID "qtrpt.project.CustomInterface")

Q\_INTERFACES(CustomInterface)

Q\_CLASSINFO("PluginName", "Plugin Name")

Q\_CLASSINFO("AddToMenu", "false")

Q\_CLASSINFO("RunOnLoading", "true")

Q\_CLASSINFO("ShowReport", "true")

public:

explicit SessionPlugin(QObject \*parent = 0);

bool *execute*(QSharedPointer<QDomDocument> xmlDoc) override;

void *saveData*(QSharedPointer<QDomDocument> xmlDoc) override;

void *showReport*(QSharedPointer<QDomDocument> xmlDoc) override;

void *clear*(QSharedPointer<QDomDocument> xmlDoc) override;

};

The following CLASSINFO parameters affect on behavior of plugin

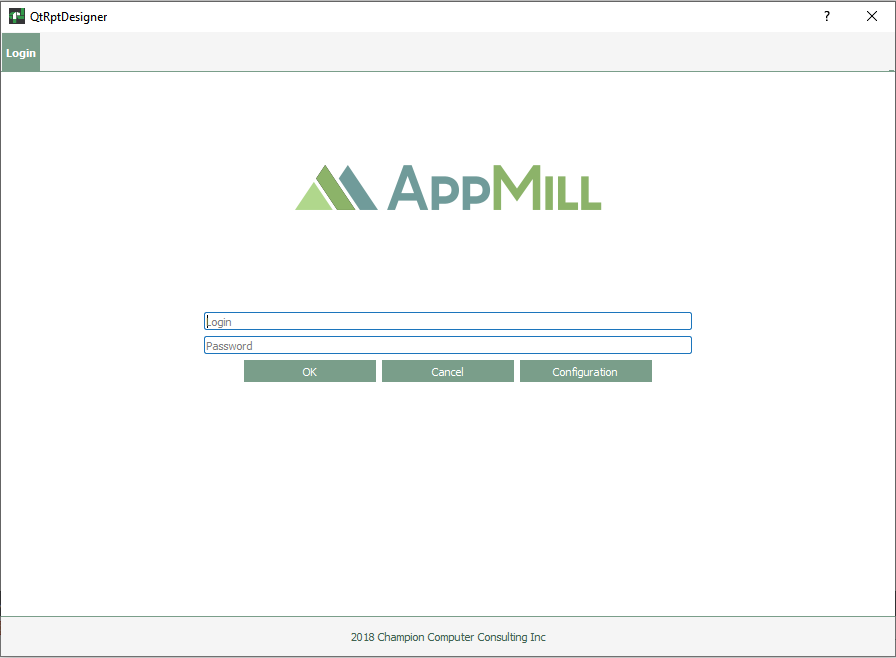
* PluginName – Just name of plugin
* AddToMenu – Define, will or not this plugin added to Menu of plugins
* RunOnLoad – Define, will or not this plugin running on start of designer
* ShowReport – Defin, will or not the plugins function “***showReport***” executes before preview of report in the designer

Description of public functions:

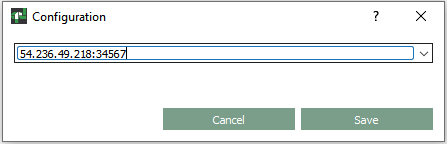
* execute – This function is calling when user select the name of plugin in Menu
* saveData – This function is calling before report saving
* showReport – This function is calling before report showing
* clear – This function is calling when the new report is created

# Plugins. Notes for AppMill users

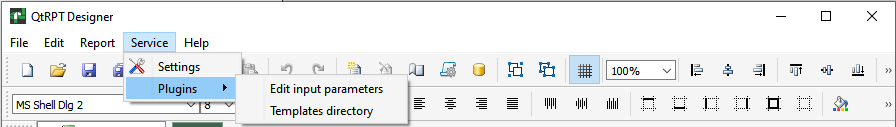
For AppMill user there a several plugins. First plugin, allow to connect and carry out verification of user’s credential at the start of the QtRptDesigner. You should enter your login, password and indicate address and port of the AppMill service. If user successful pass verification, the QtRptDesigner will be run and user will be able to send request to AppMill server and get data from it.



Before Login, user can setup the service parameters or select an AppMill service from dropdown menu or edit parameters of the connections such as IP address and port. Please note, that format as IPAdress:port, for example 54.236.49.218:34567



## Access to the plugins

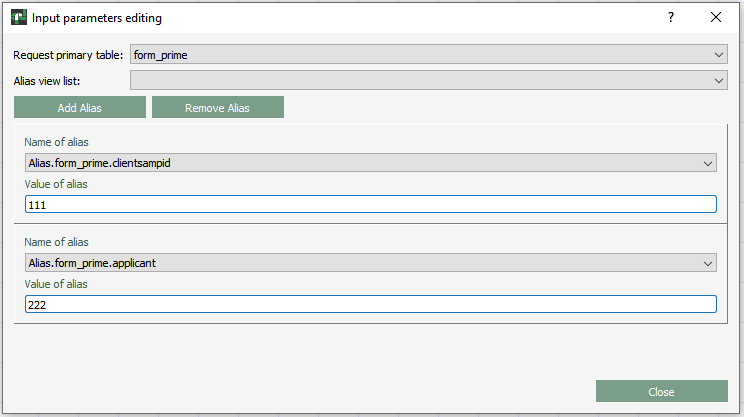


The plugins available from Main menu Service->Plugins.

## Edit input parameters plugin.

This plugin allow select the “Request primary table and “Alias view” from the list.

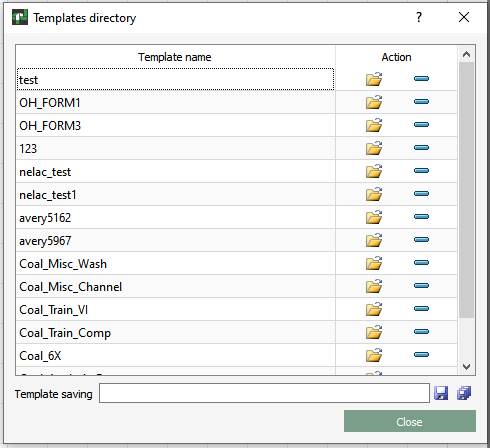
Add one or more aliases and their values. These input parameters that will be sent to the AppMill server, when user make a preview of the report.



The name of Alias can be selected from the dropdown menu. The list of aliases depends from selected Primary table of Alias View.

Each Input Alias contains two fields: Name of alias and Value of alias. To add pair, please “Add Alias” button, and enter Name and Value of Alias. To remove alias, pleas click at any field of the pair, then click at “Remove Alias” button. After editing of the Input aliases, the report (XML file) should be saved.

## Templates Directory plugin.



The Template Directory plugin allows you to load the selected report template. Use it to view data. If the report has been modified, the user has the option to save it under the same or a different name.

To download a template, select it in the list and double-click on its name or click on the button with the Folder icon.

To delete a template, select it from the list and click on the button with the Minus sign.

To save Template with the existing name, select row with the Template and click on the Save button.

To save as a new template, enter a new name and click on the Save As button.