



1 Function and area of use

The report generator allows generating Excel reports based on an Microsoft Excel made report template.

The generated report can be output to a printer or saved as an Excel file.

This document can be seen as a compliment to iX –Help documentation.

Software requirements:

- iX Developer 2.0
- Microsoft Excel or substitute

2 About this Start-Up document

This Start Up document should not be considered as a complete manual. It is an aid to be able to start up a normal application quickly and easily. For further information we refer to the manual for iX Developer 2.0. This document and other Start Up documents can be obtained from your closest distributor of operator terminals.

Please use the address *manual@beijer.se* for feedback on our Start Up documents.

Subsidiaries

3 Setting up report generator in iX

This Start-Up is based on a PC-project. While Panel-projects differs from PC-projects, the same method is used.

We will create a report that will include 3 values.

- Start iX Developer and open *Tags*.



- Create 3 tags that we will have in the report.

Tag			Controllers	
Name	Data Type	Access Right	Data Type	Controller 1
Value1	DEFAULT	ReadWrite	DEFAULT	
Value2	DEFAULT	ReadWrite	DEFAULT	
I Value3	... DEFAULT	ReadWrite	DEFAULT	

3.1 Excel template

- Create a new Excel xls-file which will be the template for iX.

The tag data to be displayed in the report file are set up in the template file using “placeholders”. The placeholders acts as substitute for the actual tag data that is to be inserted when the report is generated in iX Runtime.

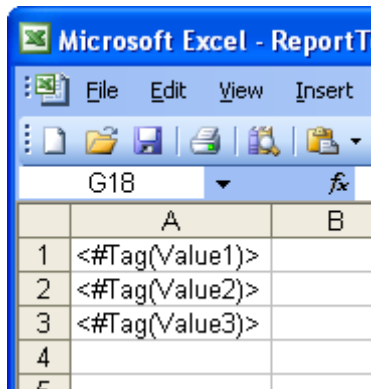
To add a placeholder, enter the following text inside a cell in the Excel report template:

<#Tag(TagName)>

Replace “TagName” with the actual name of the tag to be displayed. When the template file is complete it needs to be added to the iX Developer project.

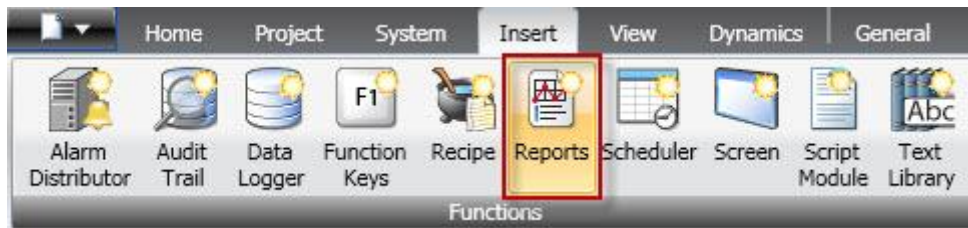
Note! The tag placeholders can only be used in the cells of the Excel template file, they can not be used in, for example, Microsoft Excel WordArt texts.

- Insert the tags (Value1,2,3) into the xml-file using the text above. Save and close.

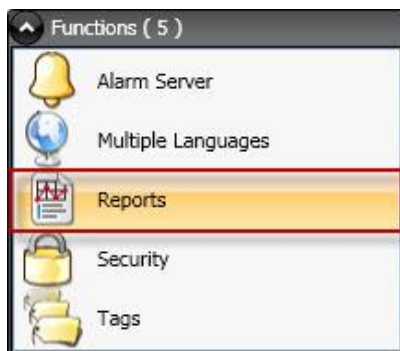


3.2 Import template in iX

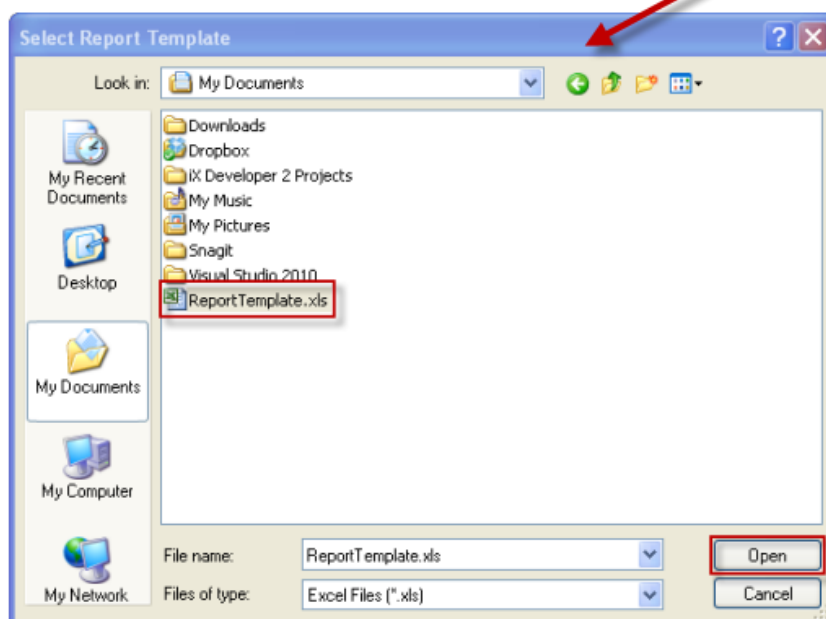
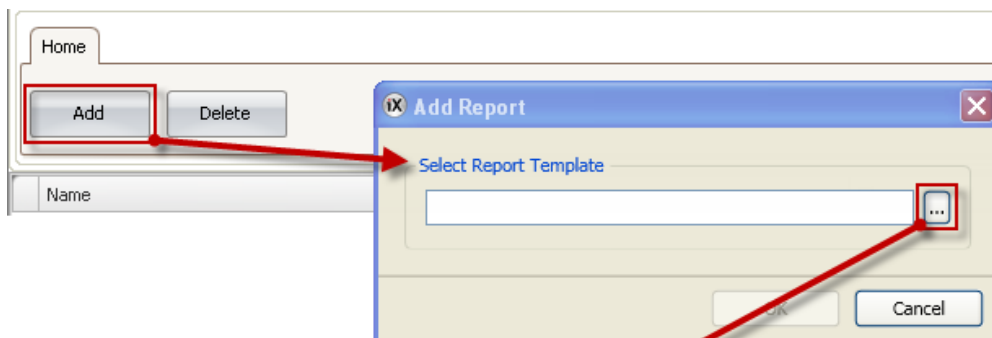
- Go back to iX Developer.
- Insert *Reports* found in the *Insert* ribbon.



- Open *Reports*.



- Add report by clicking on *Add* and browse for the template xml-file we created.

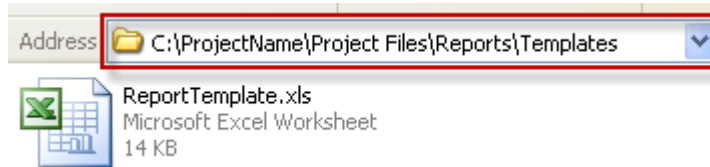


- Enter a name for the report.

Name	File name
> Report1	ReportTemplate.xls

- Save your project.

Saving the project will create a new folder structure in the project files and a copy of the template we imported.

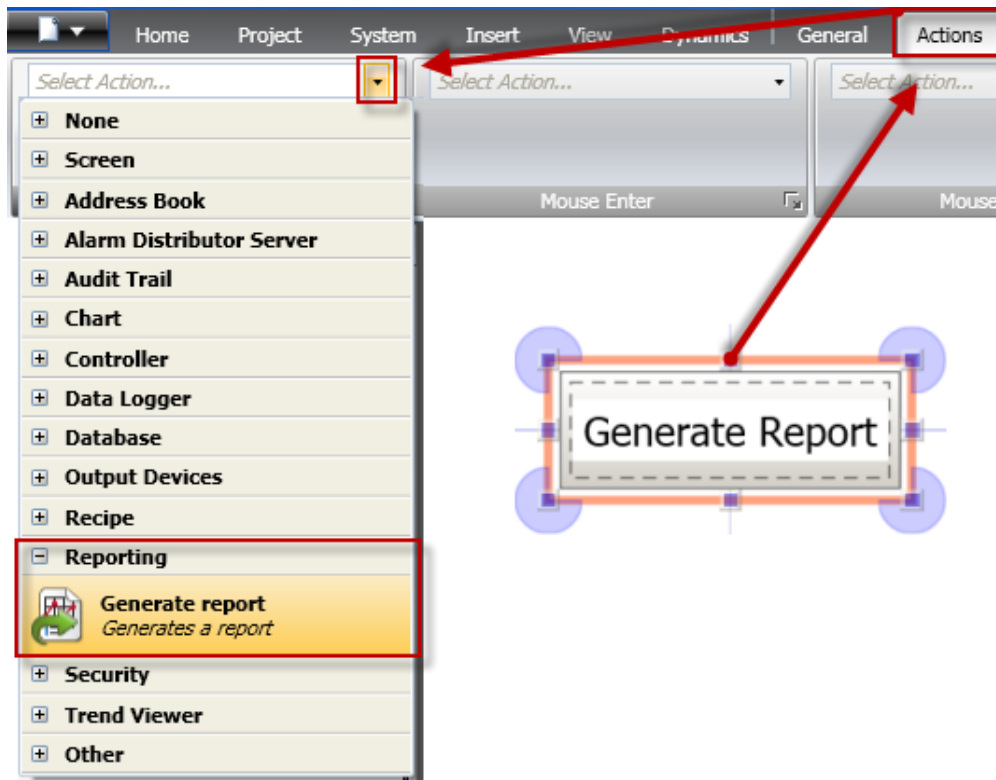


3.3 Screen layout and connections

- Create a screen layout containing of 3 analogue numerics and 1 button.
- Connect the 3 tags to the analogue numerics.

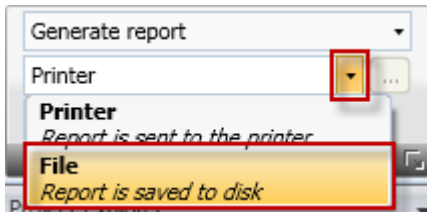


- Select the button and set action for *Click to Generate report* found under *Reporting*.

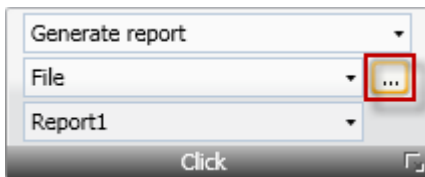


We can choose between sending the report to a printer or saving it to disk. Note: Printer setup is found under *System – Settings* in the ribbon menu.

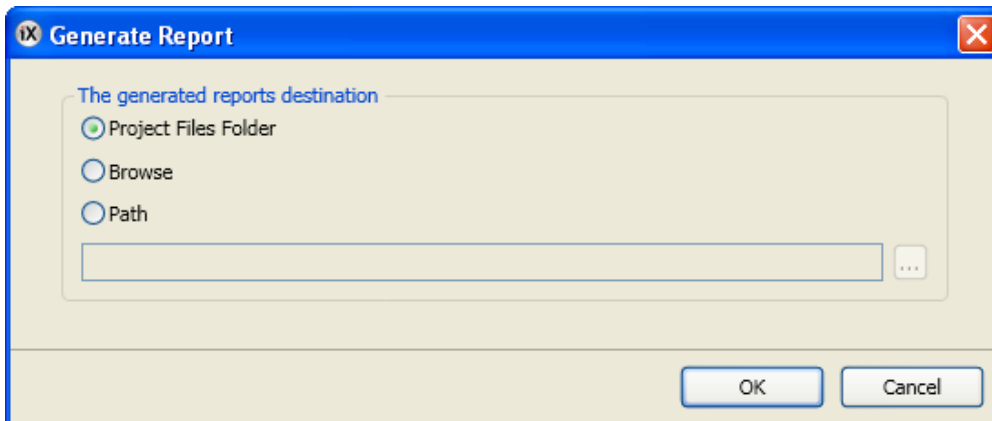
In this case select *File*.



- Having File selected, click on “...” to set the destination of the file.

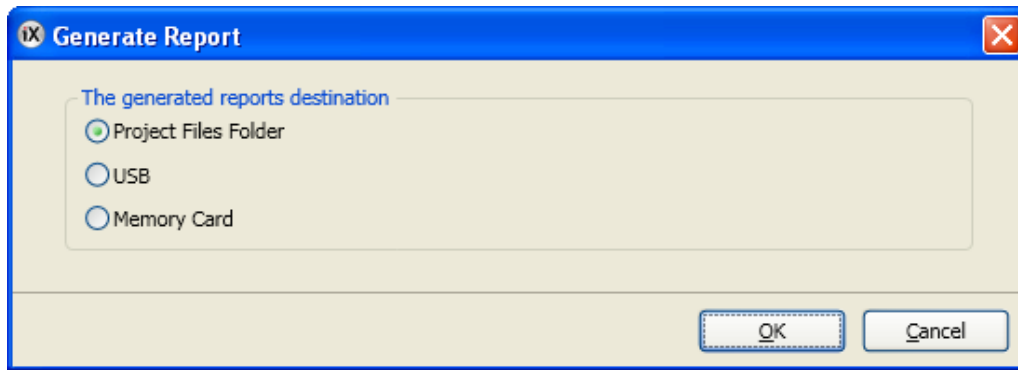


PC-project:



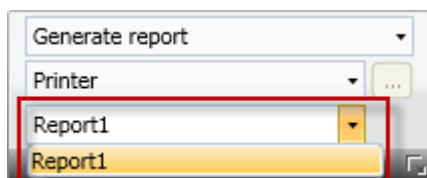
Name	Function
Project Files Folder	The file will be saved in a folder named “Project Files” in the projects execution path.
Browse	A save dialog will be show in runtime. The user will be able to select path and filename.
Path	The file will be save to a specific path.

Panel-project:



Name	Function
Project Files Folder	The file will be saved in a folder named “Project Files” in the projects execution path.
USB	The file will be saved to a connected USB memory.
Memory Card	The file will be save to a external memory card.

- In this document the file is saved to My Documents, on the PC drive, by selecting *Path*.
- Select the report you want to use. In this case only one is available.



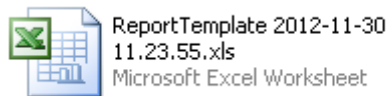
3.4 Generate report

- Run the project and enter values.
- Click on *Generate Report*.

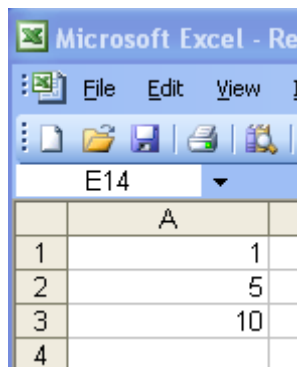
Value1: 1
Value2: 5
Value3: 10
Generate Report

The report file has now been created in the folder selected earlier. A new file will be created every time the generate button is clicked. The filename includes date and time.

- Open the file.



The values entered earlier will now be shown in the file.



Microsoft Excel - Re

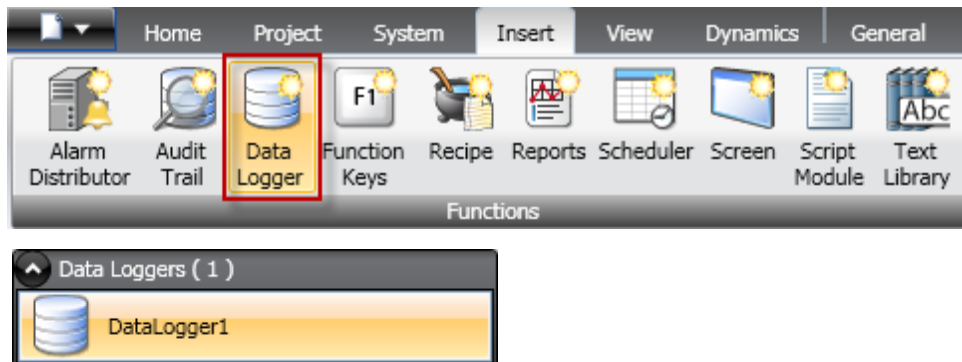
File Edit View I

E14

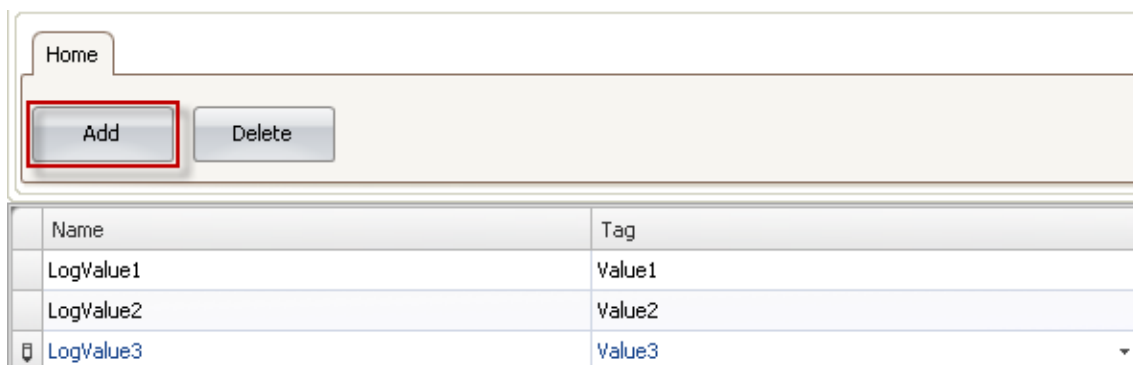
	A
1	1
2	5
3	10
4	

4 SQL

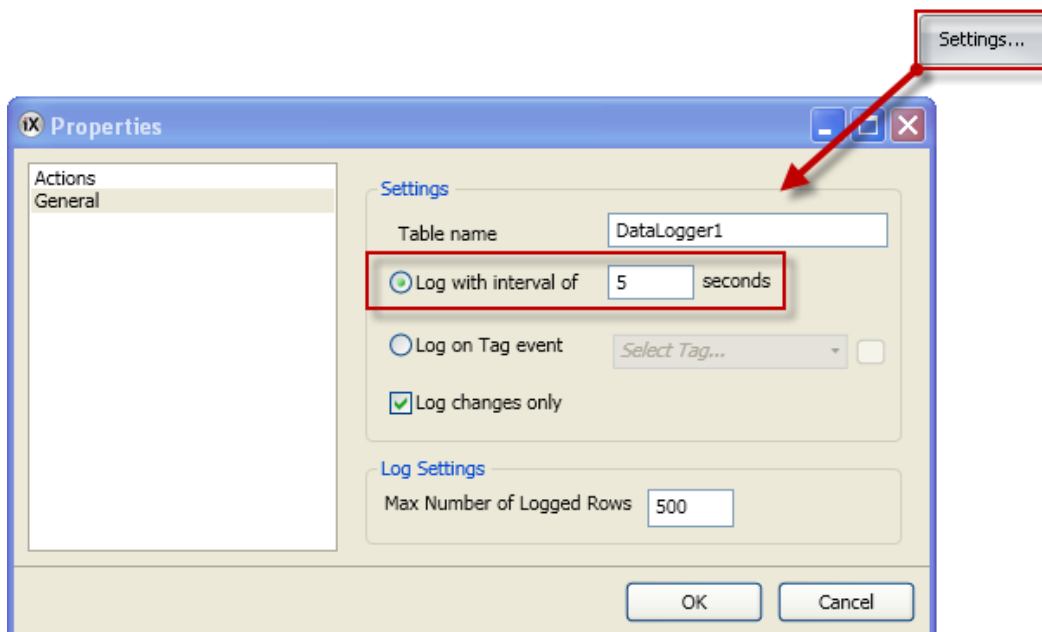
- Add Data Logger and open it.



- Add 3 log items and connect them to the tags.



- Open *Settings* and select "Log with interval of" and enter the log interval e.g. 5sec.



- Click *OK*.
- Open up the xml-template.

4.1 Configuration Sheet

The configuration sheet is a repository where configurations on the Microsoft Excel report template can be made. This sheet will not be visible in the generated report.

1. Create a new worksheet in the report template. Name the sheet <#Config>.



2. Row 10, and all rows below, in column A and B can be used to configure the SQL queries. Enter the SQL query in column B. Use the corresponding cell in column A to enter a name for the query. The name will be used to refer to the query in the report template.

Enter the SQL queries in the following format:

SQL(DatabaseName; SQL query)

In this case: SQL(General; SELECT * FROM DataLogger1)

<#Config> worksheet

	A	B	C	D
1				
2				
3				
4				
5	Row 10, and all rows below, in column A and B can be used to configure the SQL queries			
6				
7				
8	Name for the query	The SQL query		
9				
10	GetData	SQL(General;SELECT * FROM DataLogger1)		
11				

There are two available databases: General and AuditTrail.

4.2 Present the Database Data in the Report

To present the data in the report, enter the following syntax into the cells included in the named range:

<#QueryName.DatabaseColumnName>

Replace the “DatabaseColumnName” entry with the actual name of the table in the SQL database that you wish to present in the report.

6	Datalogg		
7	Value1	Value2	Value3
8	<#GetData.LogValue1>	<#GetData.LogValue2>	<#GetData.LogValue3>

The data retrieved from the SQL database, using the queries in the configuration sheet, can be presented in the report using “named ranges” in Microsoft Excel.

A named range is created in Microsoft Excel by following these steps:

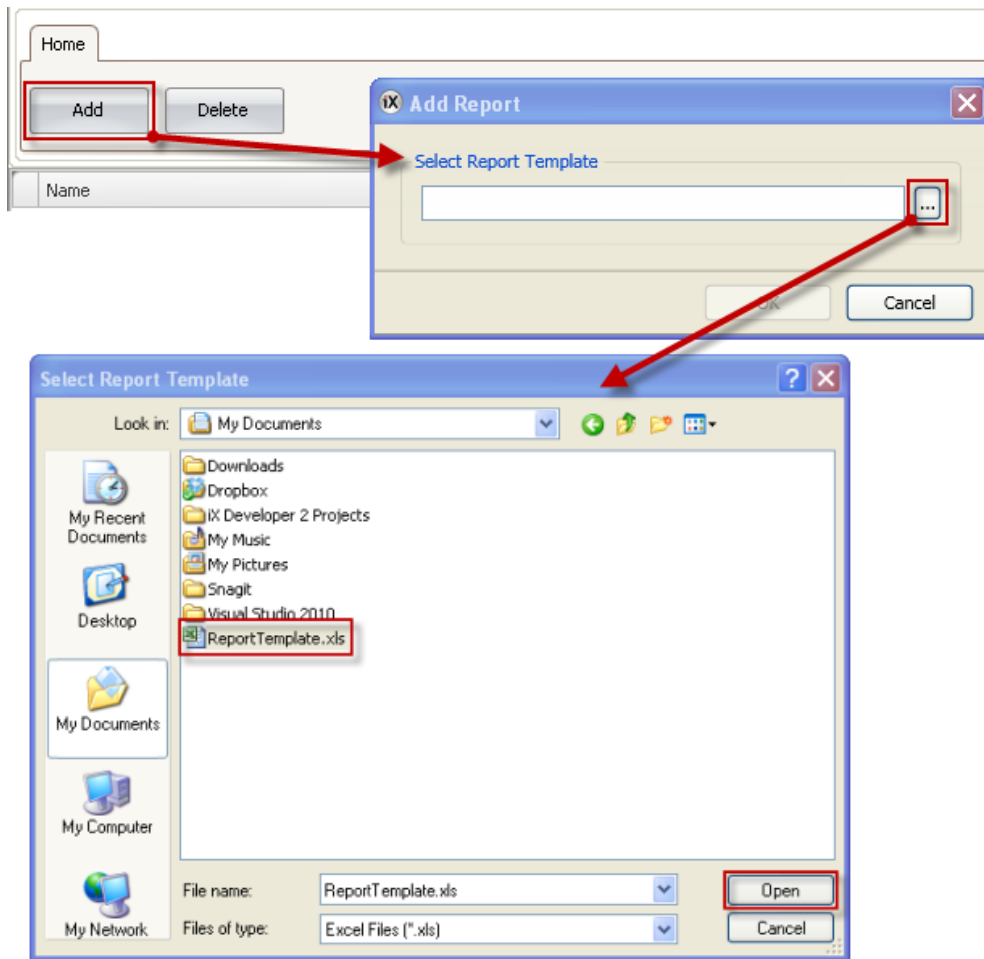
1. Select the cells to be included in a named range.
2. Click the Define name command from the Insert menu. (In Excel 2007, go to the Formulas tab and choose Name Manager).
3. Enter a name for the range. The name must be entered in the following format: “_QueryName_”. Replace “QueryName” with the name that you have chosen for your SQL query.

Primary worksheet (display values)

6	Datalogg		
7	Value1	Value2	Value3
8	<#GetData.LogValue1>	<#GetData.LogValue2>	<#GetData.LogValue3>
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			
21			
22			
23			
24			
25			

Note! It is possible to use the named range cells in Microsoft Excel charts and functions. To be able to use the named range in functions, an empty row must be inserted under the named range cells. This empty row must also be included in the function.

- Re-open the template to re-initialize the changes.



- Run the project.
- Enter random values to be logged by the Data Logger.
- Generate a report and open the report file.

You should now see the values that have been logged.

	A	B	C	D
1				
2	Value1:	51		
3	Value2:	99		
4	Value3:	425		
5				
6	Datalogg			
7	Value1	Value2	Value3	
8	3	0		0
9	13	5		7
10	23	33		7
11	33	78		45
12	43	99		1
13				

5 Limitations

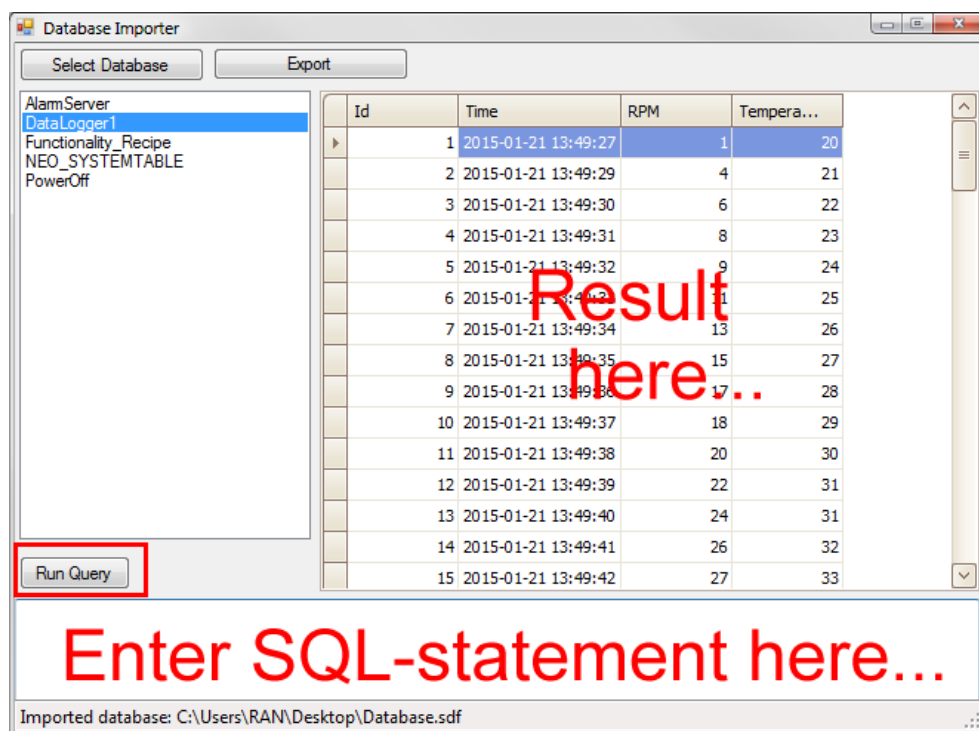
- Microsoft Excel xlsx format is not supported by the *Report Generator*.
- Reports containing charts and pictures can not be printed from a panel target.
- Reports containing macros will not be run, before being printed.
- The report generation is not an instant process. It may elapse a shorter period of time between the first sampled tag and the last sampled tag, depending on the number of tags to process.

Appendix A – Using the Database Viewer to test SQL statements

When you install iX Developer, you'll get a small helper program, the Database Viewer, installed along with iX Developer. This small application is mainly used to extract selected database-tables from a Database.sdf database backup file.

But you can use this small tool to examine and debug SQL statements.

In the SQL-statement textbox, at the bottom of the window, you can enter a SQL statement and click on [Run Query] to run the statement and display the result in the table area.



Appendix B – Useful SQL statements

In the main example in this document, we extract data from a datalogger, but you can use report generator to extract any data that is available in the database.

If you, for example would like to list everything in the alarmserver, you could use a SQL statement like this:

```
SELECT * FROM AlarmServer
```

The SQL statement above will give you everything that is available in the alarmserver, unsorted.

If you would like the result to be sorted by time and date, you could use a SQL statement like this:

```
SELECT * FROM AlarmServer ORDER BY ActiveTime ASC
```

The SQL statement above will give everything that is available in the alarmserver, sorted ascending by the time and date.

You can even make pattern matching in you statement, in the statement below only active alarms are listed:

```
SELECT TOP(20) * FROM AlarmServer WHERE State = 'Active'
```

Finally, all the statements can be concatenated together to create really advanced search and sorting functions.

If you, for example, would like the latest 20 Inactive alarms in the alarmserver sorted in a list with the oldest alarm first, you could do something like this:

```
SELECT TOP(20) * FROM AlarmServer WHERE State = 'Inactive' ORDER BY  
ActiveTime ASC
```

Note that really complex SQL statements can consume a lot of CPU power when executed. These complex statements may cause a slow panel to pause for a couple of seconds.