SQL Server Reporting Services

Microsoft's reporting tool SSRS

Johan Decorte

www.databrains.be

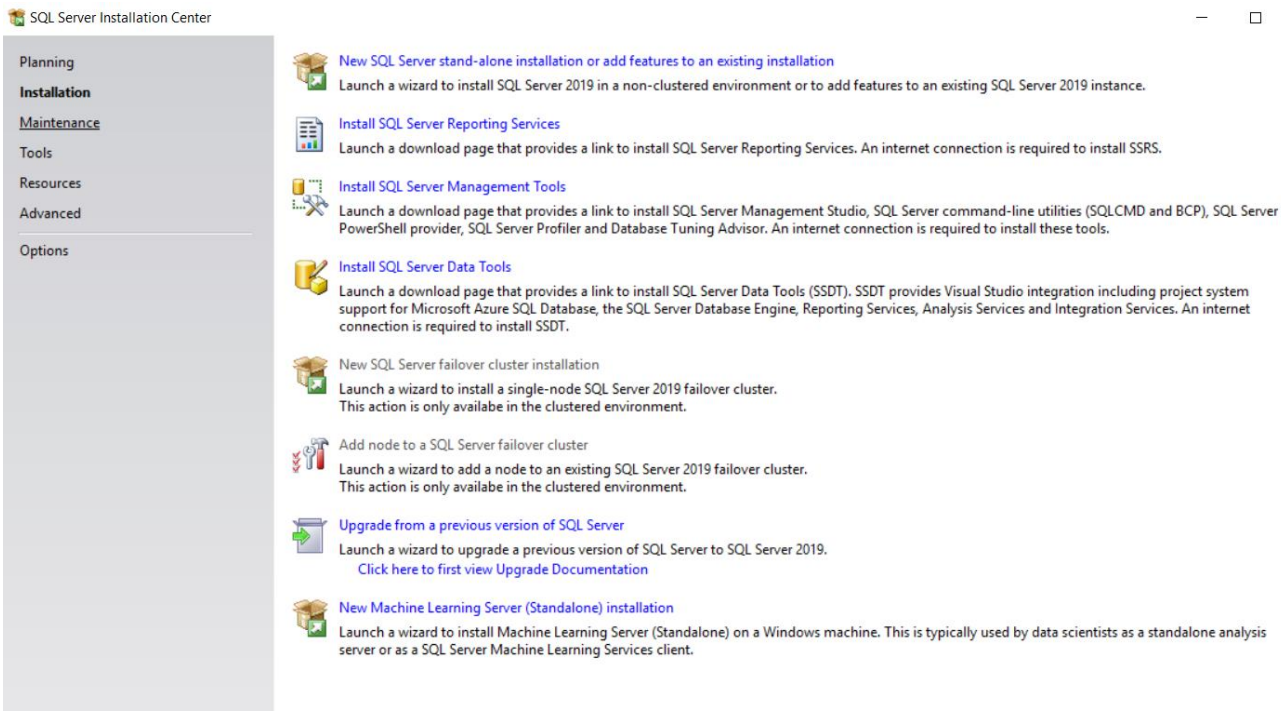
February 2023

Contents

1 Installation & Configuration

1.1 Installation

Start the SQL Server Installation Center and choose "Install SQL Server Reporting Services".



Follow the instructions on the website.

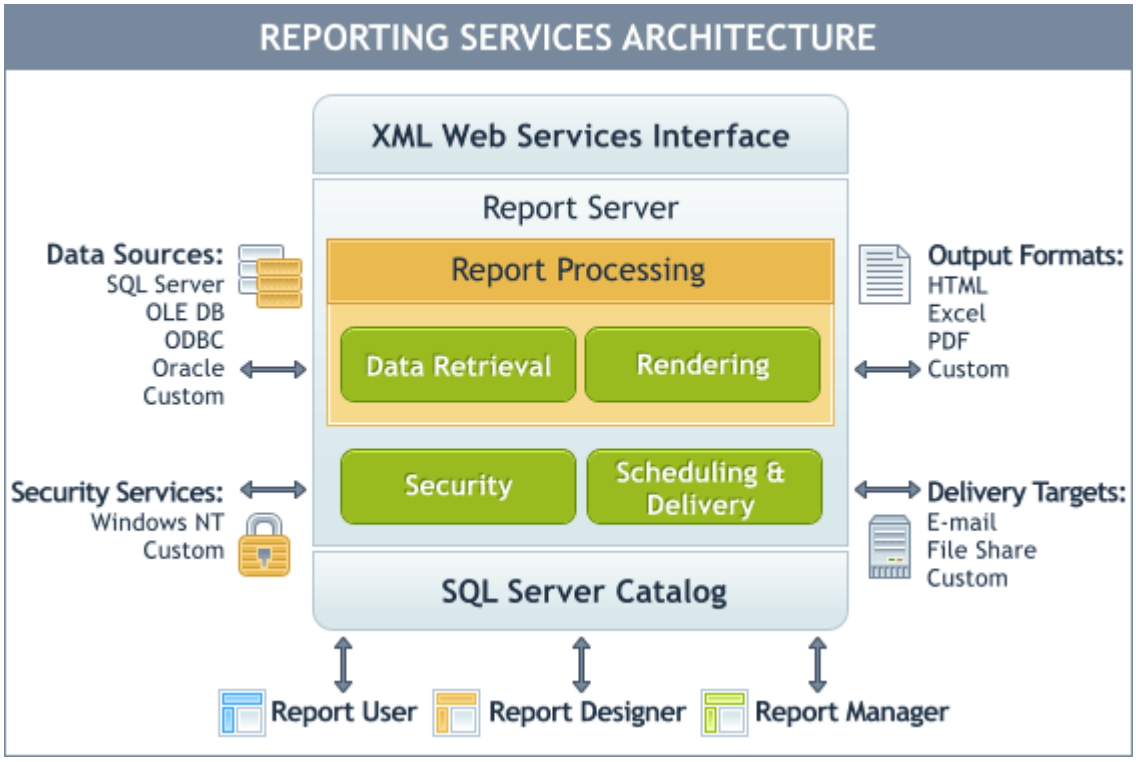
You might first have to download SQL Server Reporting Services 2019 from:

https://www.microsoft.com/en-us/download/details.aspx?id=100122

1.2 Configuration

After installation you are asked to configure Reporting Services. In section 11 we further dive into the configuration.

2 The reporting services architecture

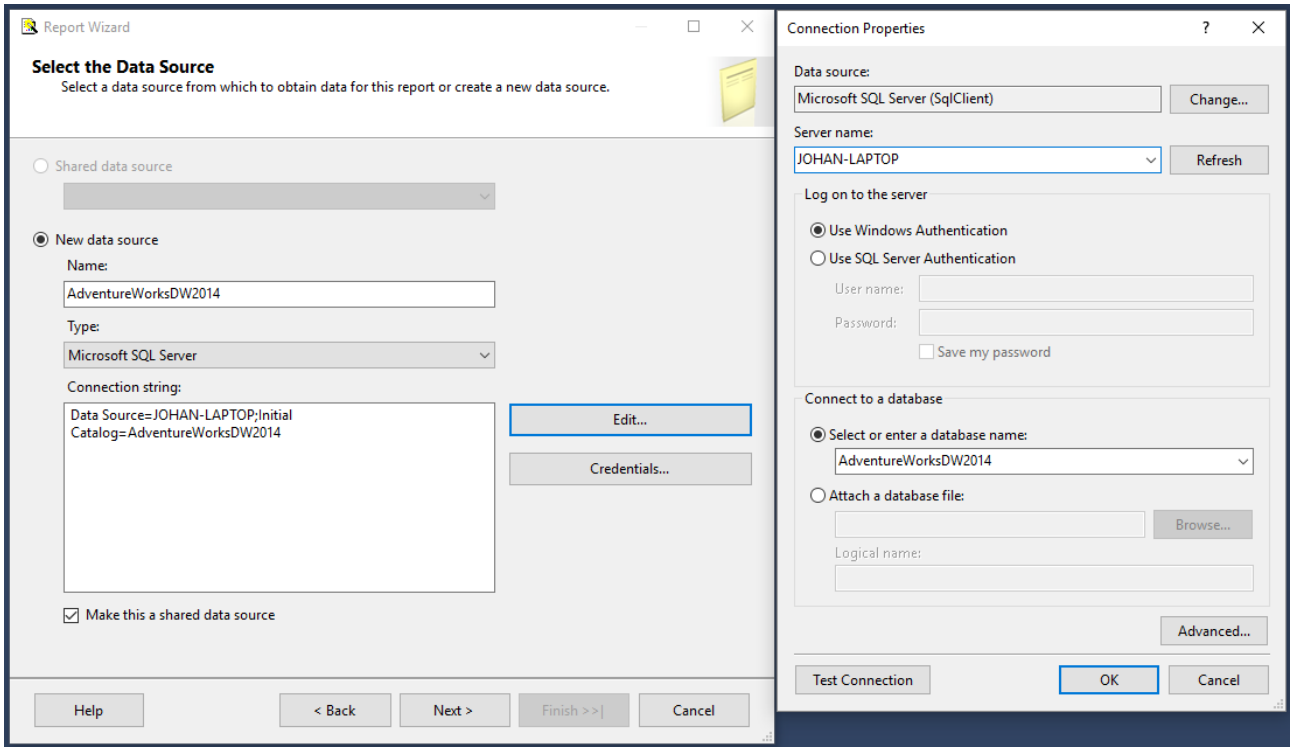


3 Developing a basic report

Designing and developing a basic report in SSRS is as easy as following the steps in a wizard. In this section, we will go through the scenario of creating our first SSRS report with the SSDT report wizard. You will learn about the structure of a Reporting Service project. Data sources will be used to connect to the source database, and then datasets will be explored which contain SQL queries to fetch data from the data source. We will go through the report layout to see how to configure the page and report layout.

In this example, you will learn how to create a report server project with SSDT. We will use the AdventureWorksDW2014 database as source. Perform the following steps to create an SSRS report:

1. Open SSDT (through Visual Studio) and create a new project. Choose a Report Server Project. Name the project Training.
2. The new project will be created with three folders: Shared Data Sources, Shared Datasets, and Reports.
3. Right-click on the Reports folder and click on the Add New Report option from the pop-up menu.
4. Now, the Report Wizard will appear. In the Select Data Source window, create a connection to the AdventureWorksDW2014 database and name the connection AdventureWorksDW2014. Check the "Make this a Shared Data Source" checkbox option. The following screenshot shows the Report Wizard and its connection properties:



5. In the Design the Query window, you can use the Query Builder option to build the query. For this example, use the following SQL code and paste it in the Query String textbox:

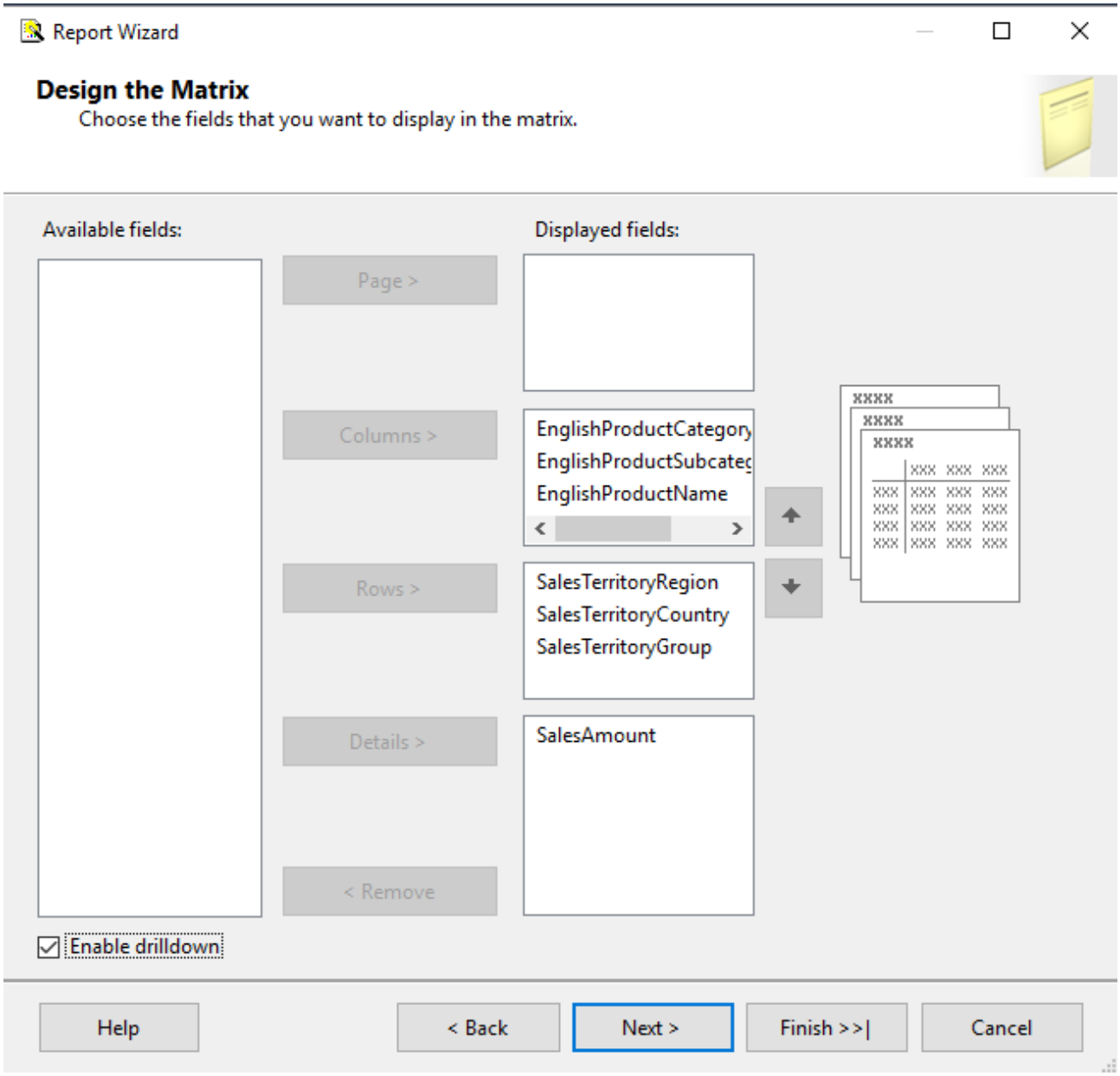
```
SELECT DimProductCategory.EnglishProductCategoryName,
DimProductSubcategory.EnglishProductSubcategoryName, DimProduct.
EnglishProductName, DimSalesTerritory.SalesTerritoryRegion,
DimSalesTerritory.SalesTerritoryCountry, DimSalesTerritory.
SalesTerritoryGroup, FactInternetSales.SalesAmount
FROM DimProduct
     INNER JOIN
     DimProductSubcategory
     ON DimProduct.ProductSubcategoryKey = DimProductSubcategory.
     ProductSubcategoryKey
     INNER JOIN
     DimProductCategory
     ON DimProductSubcategory.ProductCategoryKey = DimProductCategory.
     ProductCategoryKey
     INNER JOIN
     factInternetSales
     ON DimProduct.ProductKey = FactInternetSales.ProductKey
     INNER JOIN
     DimSalesTerritory
     ON FactInternetSales.SalesTerritoryKey = DimSalesTerritory.
     SalesTerritoryKey;
```

6. In the Select the Report Type window, choose Matrix.

7. In the "Design the Matrix" window, add EnglishProductCategoryName, EnglishProductSubcategoryName, and EnglishProductName sequentially to the Columns section. You can add them by simply dragging-and-dropping them to the columns box or by clicking on the Columns button once for each item.

8. Add SalesTerritoryGroup, SalesTerritoryCountry, and SalesTerritoryRegion respectively to the Rows section.

   Add SalesAmount to the Details section. Then, check the Enable drilldown option, as shown in the following screenshot:



9. In the last step of the wizard, rename the report to Sales by Product and Territory. Then, click on Finish to complete the wizard.
10. After finishing the wizard, you will see the report designer in SSDT with the generated report.
11. Add the Date and Time of the report generation next to the title by dragging and dropping a text field from the toolbox to the report. Write =Now() as contents of the text field.
12. Click on the Preview tab and you will see the report result. The Drilldown option is available on the report, and you can drill down with expanding groups. The following screenshot shows the Preview tab:

Sales by Product and Territory

		Accessories	Bikes			Clothing							
			Mountain Bikes	Road Bikes	Touring Bikes	Caps	Gloves			Jerseys	Shorts	Socks	Vests
							Half-Finger Gloves, L	Half-Finger Gloves, M	Half-Finger Gloves, S				
Australia		138690.6300	2853819.4486	5004548.4158	993682.1400	3811.7600	2179.6100	2791.8600	2791.8600	39660.5200	10778.4600	1006.8800	7239.0000
Canada		103377.8500	615440.4044	935616.2877	270245.7000	2175.5800	2008.1800	2008.1800	2081.6500	21877.7500	16307.6700	800.1100	5905.5000
Central		232.4600	2071.4196	539.9900					48.9800	107.9800			
France		63406.7800	899260.7108	1311933.1035	342381.9000	2130.6300	955.1100	734.7000	783.6800	15766.9500	3709.4700	287.6800	2667.0000
Germany	Germany	Europe	62232.5900	1003800.9790	1380342.8492	424370.5200	2490.2300	857.1500	563.2700	906.1300	15137.0600	629.9100	314.6500	2667.0000
Northeast			381.4200	4344.0882	1700.9900		17.9800		24.4900				63.5000
Northwest			110150.8000	1333561.5474	1716135.5138	431788.2600	2490.2300	1640.8300	2032.6700	1714.3000	27162.7400	16167.6900	925.9700	6096.0000
Southeast			531.9700	7455.8896	1565.9800	2384.0700	8.9900	24.4900			203.9600			63.5000
Southwest			145125.4200	2070024.7902	2569983.4220	858303.5700	3596.0000	2351.0400	2693.9000	2644.9200	33607.4900	20297.1000	1204.6600	8318.5000
United Kingdom			76630.0400	1162980.2866	1598217.4843	521644.8900	2966.7000	832.6600	1371.4400	979.6000	19426.2300	3429.5100	566.3700	2667.0000

4 Adding parameters to a report

One of the most vital components of reports is parameterization. Parameters help end users filter reports and select the portion of data rows that they want. In the following example, we will add the year and month parameters to the report generated in the previous section. You need to perform the following steps after the execution of the previous example to add parameters to a report:

1. Open the Sales by Product and Territory report that was created in the previous example, and in the Report Data pane, right-click on Datasets and select Add Dataset.
2. In the Dataset Properties window, rename the dataset to Years.
3. Select the Use dataset embedded in my report option. Choose Data source as AdventureWorksDW2014. Then, in the query box, type in the following query to fetch all years from the FactInternetSales table:

```
SELECT DISTINCT
OrderDateKey / 10000 AS [Year]
FROM FactInternetSales
ORDER BY 1;
```

4. Close the Dataset Properties window and then right-click on DataSet1 and select Query.
5. In the query designer window, add the following script line at the end of the SELECT query:

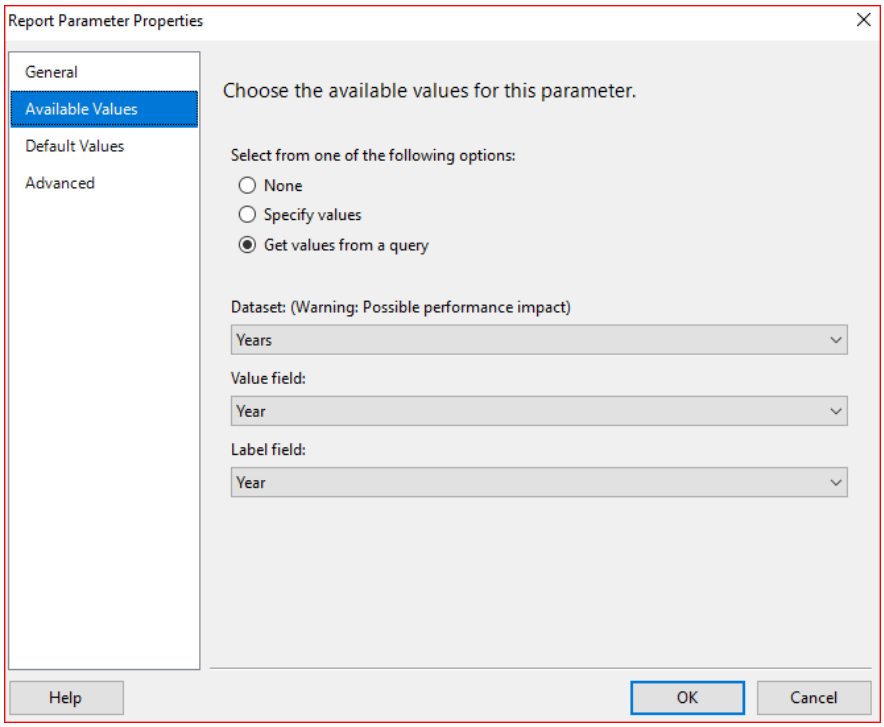
```
WHERE (FactInternetSales.OrderDateKey / 10000 = @Year)
```

6. After adding the preceding line of code, the final query looks like the following code snippet:

```
SELECT DimProductCategory.EnglishProductCategoryName,
DimProductSubcategory.EnglishProductSubcategoryName, DimProduct.
EnglishProductName, DimSalesTerritory.SalesTerritoryRegion,
DimSalesTerritory.SalesTerritoryCountry, DimSalesTerritory.
SalesTerritoryGroup, FactInternetSales.SalesAmount
FROM DimProduct
      INNER JOIN
      DimProductSubcategory
      ON DimProduct.ProductSubcategoryKey = DimProductSubcategory.
      ProductSubcategoryKey
      INNER JOIN
      DimProductCategory
      ON DimProductSubcategory.ProductCategoryKey = DimProductCategory.
      ProductCategoryKey
      INNER JOIN
      factInternetSales
      ON DimProduct.ProductKey = FactInternetSales.ProductKey
      INNER JOIN
```

```
    DimSalesTerritory
    ON FactInternetSales.SalesTerritoryKey = DimSalesTerritory.
       SalesTerritoryKey
WHERE (FactInternetSales.OrderDateKey / 10000 = @Year)
```

7. Close the Query window and go to the Preview window. You will see that the report asks for the value for Year in a textbox. Enter 2012 and click on View Report. Change the year and you will see that the report's data will change.
8. Now we want to change the user interface for the Year filter to be a drop-down list.
9. Go back to the Report designer, expand the Parameters folder in the Report Data pane, and double-click on the Year parameter.
10. Leave the General tab as is and go to Available Values. Choose the Get values from a query option, as shown in the following screenshot:



11. Choose Dataset as Years and select Year for both the Value and Data fields.
12. Close the Parameter window and go to the Preview pane. You will see that the textbox filter for Year changes to a drop-down list and you can see a list of years and can select any of them.
13. Now we want to add the Month filter. The month will be selected after the year as a cascade filter.
14. In SQL Server Management Studio create a view from the report query but add the OrderdateKey field:

```
create view report1 as
SELECT OrderdateKey,DimProductCategory.EnglishProductCategoryName,
DimProductSubcategory.EnglishProductSubcategoryName, DimProduct.
EnglishProductName, DimSalesTerritory.SalesTerritoryRegion,
DimSalesTerritory.SalesTerritoryCountry, DimSalesTerritory.
SalesTerritoryGroup, FactInternetSales.SalesAmount
FROM DimProduct
     INNER JOIN
     DimProductSubcategory
     ON DimProduct.ProductSubcategoryKey = DimProductSubcategory.
     ProductSubcategoryKey
```

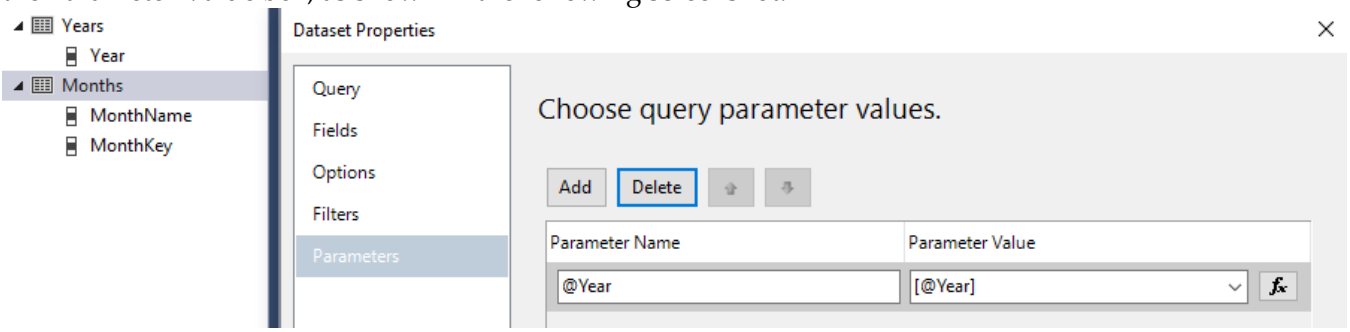
```
INNER JOIN
DimProductCategory
ON DimProductSubcategory.ProductCategoryKey = DimProductCategory.
ProductCategoryKey
INNER JOIN
factInternetSales
ON DimProduct.ProductKey = FactInternetSales.ProductKey
INNER JOIN
DimSalesTerritory
ON FactInternetSales.SalesTerritoryKey = DimSalesTerritory.
SalesTerritoryKey;
```

15. Go back to the report designer and right-click on the Datasets folder in the Report Data pane and add a new dataset.

16. Rename the new dataset to Months and choose the embedded dataset option. Select the data source and write the following query in the Query box:

```
SELECT DISTINCT
     EnglishMonthName AS MonthName,
     MonthNumberOfYear AS MonthKey
FROM report1 r join dimdate d on r.orderdatekey=d.DateKey
WHERE OrderDateKey / 10000 = @Year
ORDER BY 2;
```

17. Go to the Parameters tab of the Dataset Properties window and choose the [@ Year] parameter in the Parameter Value box, as shown in the following screenshot:



18. Close the Dataset Properties window and edit the query for DataSet1 by adding the following line in the query:

    AND ((FactInternetSales.OrderDateKey / 100 % 100) IN (@Month))

19. After adding the preceding line of code, the following will be the final query of DataSet1:

```
SELECT DimProductCategory.EnglishProductCategoryName,
DimProductSubcategory.EnglishProductSubcategoryName, DimProduct.
EnglishProductName, DimSalesTerritory.SalesTerritoryRegion,
DimSalesTerritory.SalesTerritoryCountry, DimSalesTerritory.
SalesTerritoryGroup, FactInternetSales.SalesAmount
FROM DimProduct
     INNER JOIN
     DimProductSubcategory
     ON DimProduct.ProductSubcategoryKey = DimProductSubcategory.
     ProductSubcategoryKey
     INNER JOIN
     DimProductCategory
           ON DimProductSubcategory.ProductCategoryKey = DimProductCategory.
     ProductCategoryKey
     INNER JOIN
     factInternetSales
     ON DimProduct.ProductKey = FactInternetSales.ProductKey
```

```
    INNER JOIN
    DimSalesTerritory
    ON FactInternetSales.SalesTerritoryKey = DimSalesTerritory.
    SalesTerritoryKey
WHERE FactInternetSales.OrderDateKey / 10000 = @Year
      AND (FactInternetSales.OrderDateKey / 100 % 100) IN (@Month)
```

20. Under the Parameters folder in the Report Data pane, double-click on the Month parameter to edit.
21. In the General tab, check the option Allow multiple values.
22. In the Available Values menu, choose Get Values from a query. Next, choose the Months dataset, select MonthKey as the Value field, and select MonthName as the Label field.
23. Preview the report and select the year. You will now be able to choose the month. Note that you can choose multiple months, as shown in the following screenshot:

Sales by Product and Territory

	⊞ Accessories	⊞ Bikes	⊞ Clothing
⊞ Australia	22057.9100	663621.1500	11358.6600
⊞ Canada	15370.4200	144382.5200	7317.7900
⊞ Central	21.2600	539.9900	53.9900
⊞ France	10539.2300	244335.7000	4766.8000
⊞ Germany	9870.1300	260162.3600	3675.3100
⊞ Northeast	149.9600		24.4900
⊞ Northwest	18952.8300	344501.1700	9170.4100
⊞ Southeast	4.9900		24.4900
⊞ Southwest	24094.1700	516428.0600	12270.0100
⊞ United Kingdom	13187.7500	346158.7500	6160.2500

5 Printing and page configuration

Reporting Services reports can be designed in a printer-friendly manner. There are properties in reports that are helpful in the configuration of the page size for printing. In this section, we will go through some page configuration options for the report.

1. Create a new report and use the same data source for it, AdventureWorksDW2014. Then, use the following query as the query string:

```
SELECT FirstName, LastName, BirthDate, MaritalStatus, Gender,
TotalChildren, NumberChildrenAtHome, EnglishEducation, NumberCarsOwned
FROM DimCustomer;
```

2. Choose Tabular as the report type. Next, select all columns to be in the Details section in the Design the Table window. Rename the report to Customers and complete the wizard.
3. Preview the report. In the Preview window, click on the Export button and export the report to PDF.
4. Save the PDF file and then open it. You will notice that the number of pages is 1608 by default. Next, the pages exported to the portrait layout caused each report page to split into two PDF pages.

5. Go back to the report designer. In the Properties window, select the Report object. Then, expand the PageSize properties and change Height to 29,7cm and Width to 21 cm. Also, change all margins to 1cm, as shown in the following screenshot:

⊟ Margins	1cm; 1cm; 1cm; 1cm
Left	1cm
Right	1cm
Top	1cm
Bottom	1cm
⊟ PageSize	29,7cm; 21cm
Width	29,7cm
Height	21cm

6. Rearrange the column width so data is never splitted over two lines.
7. Right mouse-click on the [BirthDate] field, click Customer in the Placeholder Properties Window and choose "dd-MMM-yyyy" as Custom format.
8. Save the changes and preview the report. Export the report to PDF. This time, you will see that the number of pages is 552. Each report page is exported in portrait in one PDF page.

6 Sorting and grouping

In the following example, we will modify the customers' report generated in the previous example. We will add a SalesAmount column for each customer. We will apply sorting on the report and we will add a group for NumberChildrenAtHome. We will also add a subtotal for each group item. Perform the following steps:

1. In the Customers report, go to the Report Data pane and change the query of DataSet1 to the following script:

```
SELECT FirstName, LastName, BirthDate, MaritalStatus, Gender,
TotalChildren, NumberChildrenAtHome, EnglishEducation, NumberCarsOwned,
SUM(FactInternetSales.SalesAmount) AS SalesAmount
FROM DimCustomer
     LEFT OUTER JOIN
     FactInternetSales
     ON DimCustomer.CustomerKey = FactInternetSales.CustomerKey
GROUP BY FirstName, LastName, BirthDate, MaritalStatus, Gender,TotalChildren,
NumberChildrenAtHome, EnglishEducation, NumberCarsOwned;
```

2. Now you will see the SalesAmount column under DataSet1. Drag-and-drop this column to the report designer exactly after the last column in the report.
3. Set the Report Property "Language" to "nl-NL" or "=user!Language" (to force to regional settings of user). Select the [SalesAmount] field in the report designer. Then, in the Properties window, change the Format property to # #,##€.
4. Select one of the column headers in the table in the report designer. Then, right-click on Table Header Columns and choose Tablix Properties... from the pop-up menu.
5. In the Tablix Properties… window, go to the Sorting tab. Then, click on the Add button and choose the SalesAmount column with descending order (Z to A).

6. Preview the report. You will see that the SalesAmount column will be shown with a thousand separator, a decimal comma, and a euro sign. The report is ordered descending by sales amount.
7. Now we want to add interactive reporting to column headers.
8. Click on the LastName column header textbox (select the textbox itself – not the text in it) and then right-click on it and choose Textbox Properties from the pop-up menu.
9. In the Textbox Properties window, go to the Interactive Sorting tab, check Enable interactive sorting on this text box, and choose the LastName column in the Sort by section.
10. Preview the report. Now, you can click on the LastName column to change the sorting of rows (ascending or descending) based on this column.
11. Go back to the report designer. Click on the second row (the row after the column headers) and then right-click on the header of that row. Then, from the pop-up menu, select Add Group. Next, click on Parent Group.
12. In the Tablix group window, select the [NumberChildrenAtHome] column in the Group by drop-down list and check the Add group header option.
13. You will see that a new row has been added to the Tablix between header and Details rows. This row is a group header row. Click on the textbox for the SalesAmount column in the group header row and then choose the SalesAmount expression from the pop-up menu.
14. In the Properties window for this textbox, choose the appropriate format.
15. Preview the report. You will see the new group for NumberOfChildrenAtHome. The group header row shows the total sales amount for that group.
16. Now, we want to show the average number of cars for each group in the group header.
17. Right-click on the textbox for the NumberCarsOwned column in the group header row and select Expression.
18. In the Expression window, enter the following script: =Avg(Fields!NumberCarsOwned.Value)
19. Close the Expression window. Change the Format property of this textbox to an appropriate value.
20. Preview the report and you will see an average number of cars shown in the group header.

7 Expressions

SSRS expressions play a vital role in creating calculated fields in the report. Expressions are also an important part of dynamism in reports; for example, the background colour of text can be changed based on an expression. In this section, we will have a quick overview of expressions with an example that is based on the previous reporting project. In the following example, we want to distinguish records for male and female customers by changing the background colours of the row. For this purpose, we will write an expression and use that in the background colour property of the data row using the following steps:

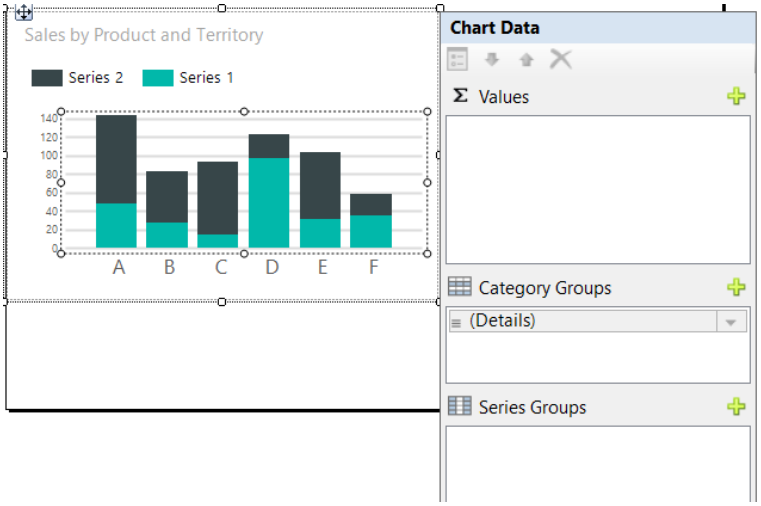
1. Go to the report designer of the Customer report from the previous example.
2. Click on the row header for the details record (third row in the Tablix).

3. In the Properties window, find the Background Colour property and click on your desired colour from the color picker menu.
4. Choose Expressions from the colour picker pop up.
5. In the Expression editor window, enter the following script: =iif(Fields!Gender.Value="M","PaleTurquoise","Khaki"). You can also refer to the current field with *me.value*.
6. Close the Expression window and preview the report. You will see that male and female data rows are now different with regard to background colour.

8 Adding charts

Charts, KPIs, and dashboards are some of the main components of analytical reporting. Reporting Services uses a wide range of charts and KPIs with highly configurable settings that provide a robust reporting platform for the end user. In the next example, we will add a chart for the sales information in the **Sales by Product and Territory** report from the first example. We will just create a stacked column chart with the default configuration from the existing dataset with the help of the following steps:

1. Open the report designer for the Sales by Product and Territory report.
2. Click on an empty area in the report designer (inside the rectangle surrounding the report) and then go to the Properties window and choose the Body object.
3. Change the size of the Body object with these parameters: Width to 12 cm and Height to 12 cm.
4. Drag-and-drop a Chart object from Toolbox into the report designer under the matrix object.
5. Choose Chart type as Stacked Column.
6. Change the title of the chart to Sales by Product and Territory.
7. Click on the chart shape and you will see the Chart Data pane appear on the right-hand side of the chart, as shown in the following screenshot:



8. From the Report Data pane, drag-and-drop SalesAmount into the Values section of Chart Data. Drag-and-drop EnglishProductCategoryName into Series Groups.

9. Add execution time to the lower right corner of the report by drag-and-dropping from the Report Data menu the built-in field "Execution Time". Set language in the properties window to "nl-NL".
10. Preview the report for all months of the year 2013 and the month March and you will see that information in the chart right below the matrix.

9 Use Report Builder as an alternative for SSDT

Report builder is a standalone app, that can replace SSDT for creating reports. If installed on the deployment server, it can also be used to edit reports directly from the Report Server.

Download from https://www.microsoft.com/en-us/download/details.aspx?id=53613

10 Improve the presentation

10.1 Include header and footer details

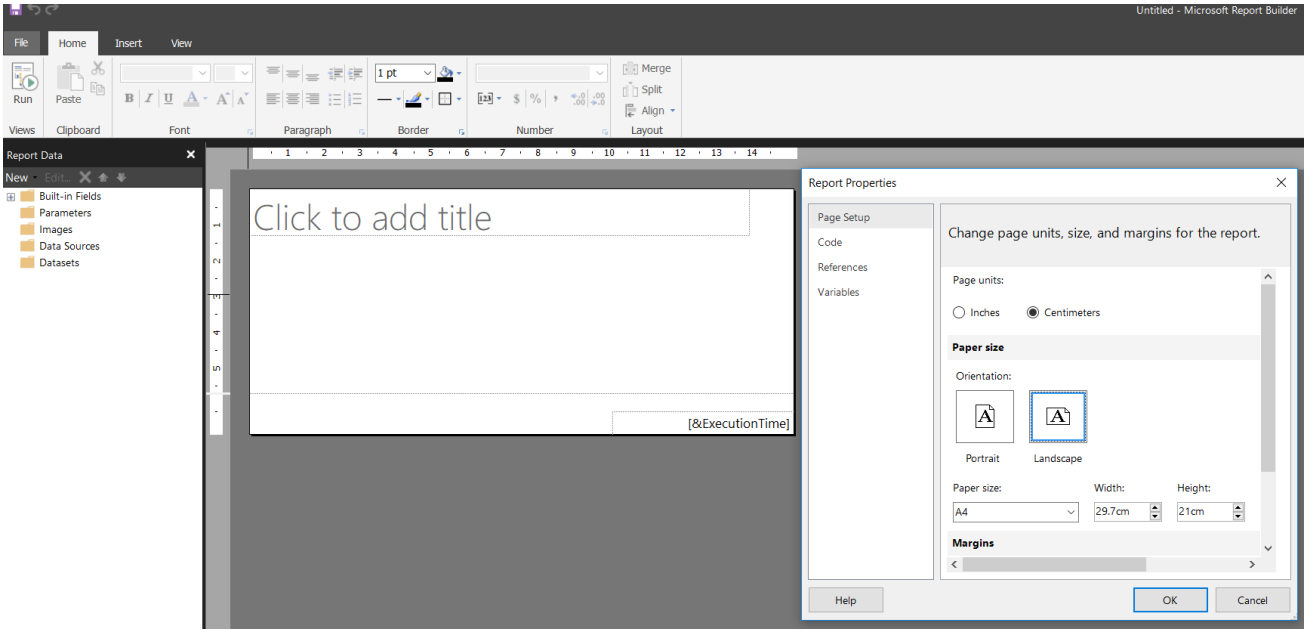
For printed reports headers and footers can be necessary.

In SSDT create a new shared dateset based on this query (datasource AdventureWorks2014DW):

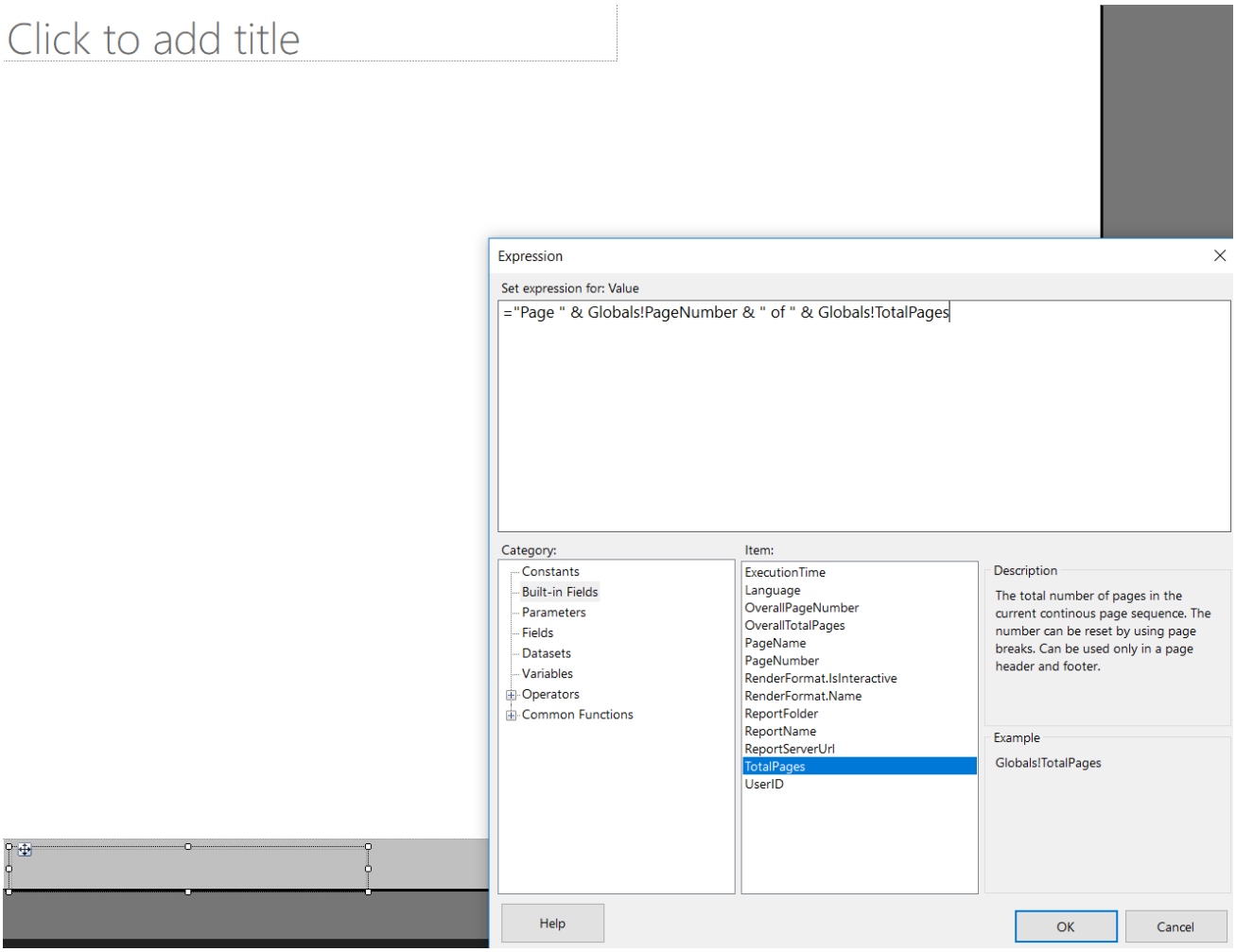
```
select pc.EnglishProductCategoryName,g.City,SUM(salesamount) Sales
from FactInternetSales f
join DimCustomer c
on f.CustomerKey=c.CustomerKey
join DimGeography g on c.GeographyKey=g.GeographyKey
join DimProduct p on f.ProductKey=p.ProductKey
join DimProductSubcategory psc on p.ProductSubcategoryKey=psc.ProductSubcategoryKey
join DimProductCategory pc on psc.ProductCategoryKey=pc.ProductCategoryKey
roup by pc.EnglishProductCategoryName,g.City
```

and deploy it.

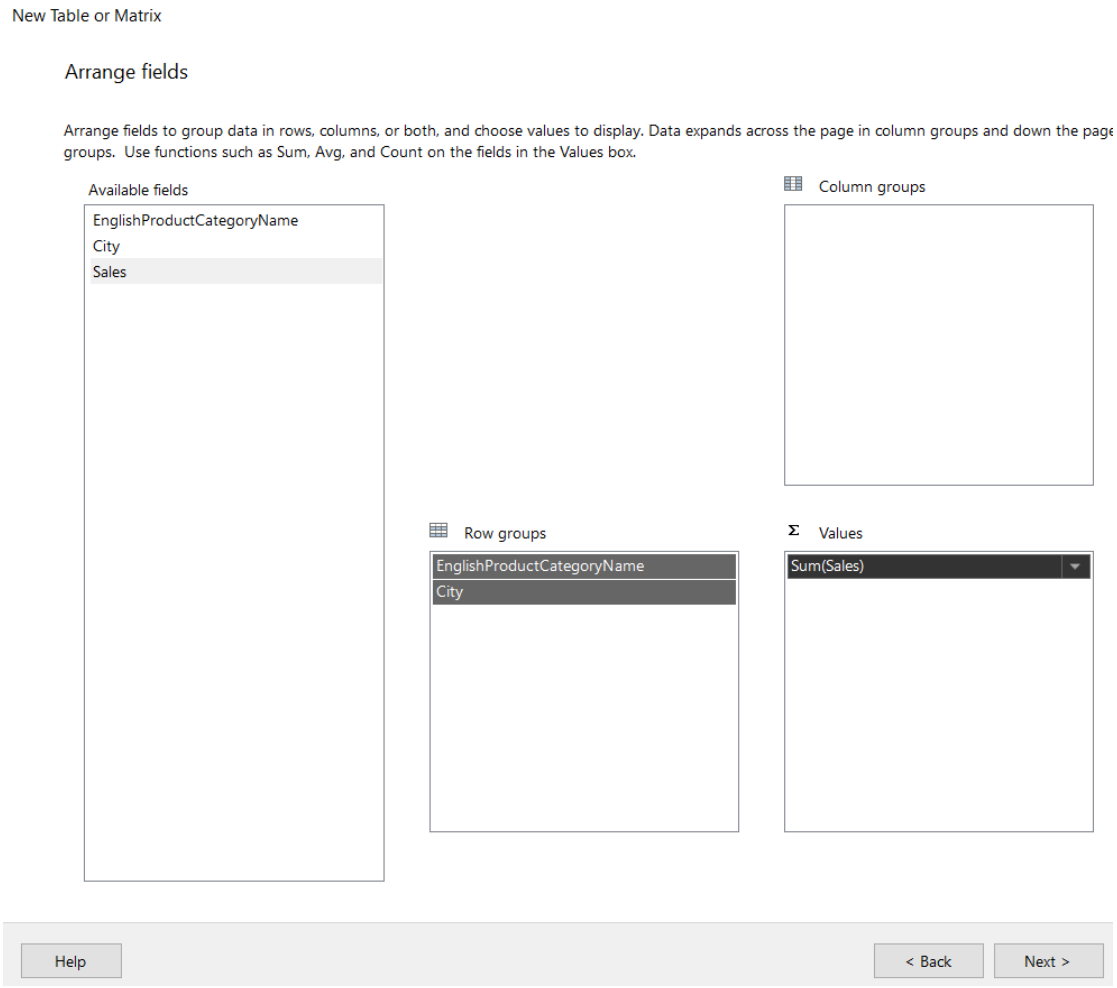
Now from the Report Builder create a new report and set the paper size to A4/landscape:



Now click on the footer, change the back ground colour to grey and show the page number and total number of pages:



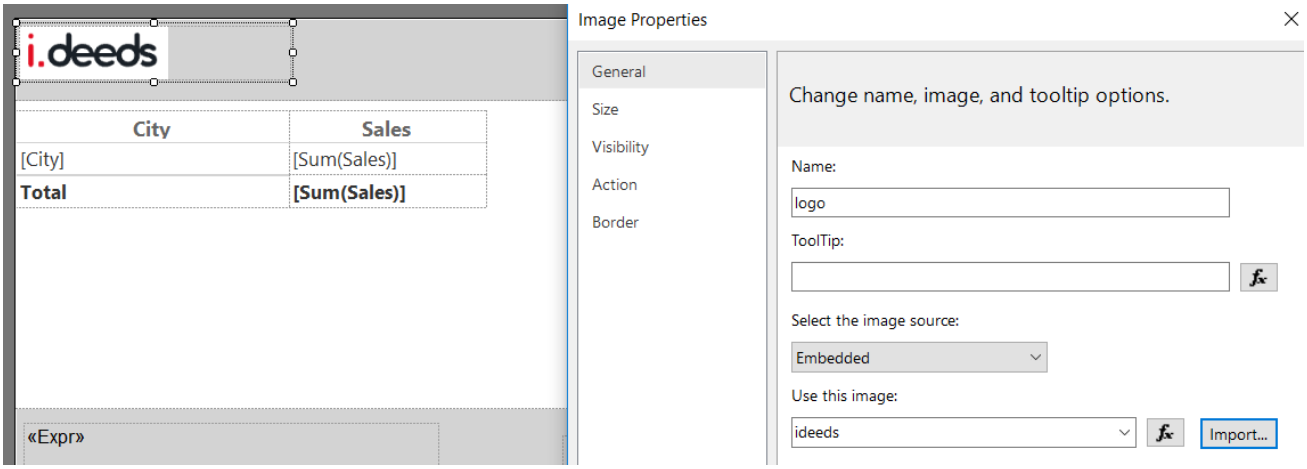
Add a quick table by using the menu Insert / Table / Table Wizard.



Set the visibility of the details line to "Show":

10.2 Add graphics and logos

You can add an image by dragging the image icon (from the Insert menu) to e.g. your header and importing an image file:

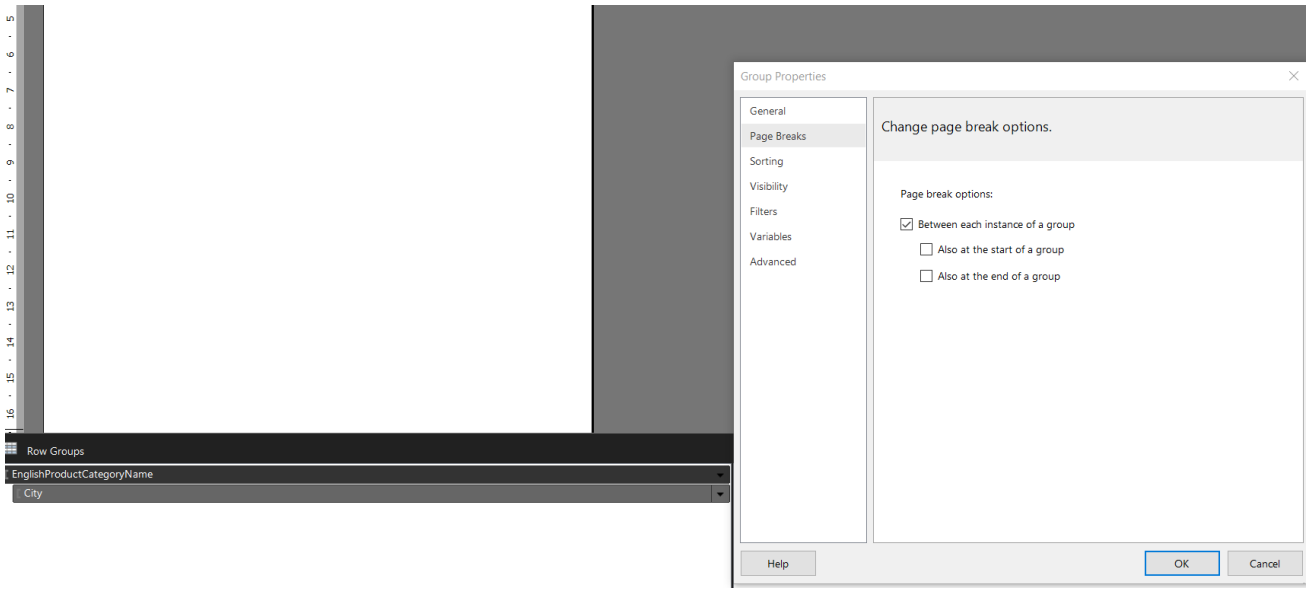


10.3 Align page elements

You can align e.g. logo and the title in the header using the Align menu:

10.4 Add page breaks

Open the Group Properties dialog box from the EnglishProductCategoryName Row Group and in the Page Break section check "Between each instance of a group".



11 Exercise Report Creation

Use the xtreme database for following exercise.

Create, in SQL Server Management Studio a view that shows the total sales (from orders.orderamount) per year and customer country (see extract below). Create a matrix report that uses this view. Determine yourself where you can best put the three fields Year, Country and Sales.

Year	Country	Sales
2019	Austria	16180.28
2019	Belgium	3582.37
2019	Canada	12065.09
2019	England	3014.15
2019	France	106.30
2019	Italy	13369.21
2019	Mexico	764.85
2019	Netherlands	6811.52
2019	New Zealand	1778.90
2019	Norway	1799.70
2019	Poland	1973.65
2019	Portugal	719.46
2019	South Africa	1115.30
2019	Spain	11379.21
2019	Sweden	4024.35
2019	Switzerland	9487.15
2019	USA	171220.15
2020	Argentina	1664.70
2020	Aruba	5879.70
2020	Australia	9899.99
2020	Austria	10304.63
2020	Bahamas	659.70
2020	Bangladesh	65.70
2020	Barbados	329.85
2020	Belgium	41737.62

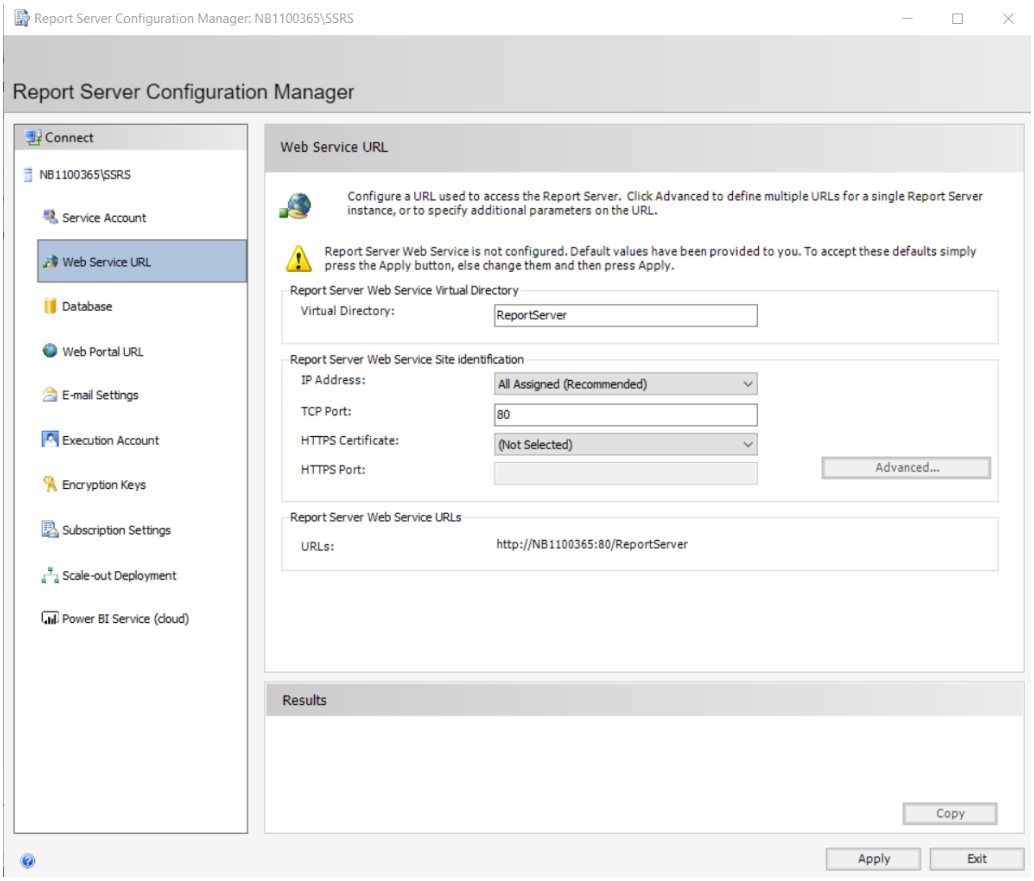
12 Deploying and configuring reports

Developed reports need to be deployed to UAT (user acceptance test) or production environments. SSRS reports can be deployed easily from SSDT. Reports can also be deployed from the WebUI to the Report Manager. There are two web applications for Reporting Services: Report Server for deployment and browsing reports and Report Manager for configuration, security, and administration tasks on reports and data sources. There is a tool for configuring SSRS components, named SQL Server Reporting Services Configuration Manager. In this section, we will deploy our reports into Report Server and then we will have a very quick look at Report Manager and Configuration Manager.

In this example, we will first find the report server URL from Reporting Services Configuration Manager, and then we will use that URL to deploy reports from SSDT. Finally, we will browse those reports to see them from the Report Server web application by using the following steps:

1. Go to Start | Microsoft SQL Server 2019 Reporting Services Configuration Manager.
2. When Reporting Services Configuration Manager opens, it may ask for a connection. Just choose the default instance and connect.

3. You will see the service status that shows information about the SQL Server version, edition, and service status information.
4. Click on Web Service URL. The URL of the Report Server web application can be viewed or changed in this option, as shown in the following screenshot:



5. Go to SSDT and right-click on the Reporting Service project, and then from the pop-up menu, select Properties.
6. In the Project Properties window, change the TargetServerURL option to the URL of the report server that you picked from Reporting Services Configuration Manager. Then, close the Properties window.
7. Right-click on the project and click on Deploy to deploy the project, as shown in the following screenshot:
8. After successful deployment, go to the report server URL on your browser. In the report server web page, open the Training folder and navigate to any of the reports there.