SQL Server 2012 Tutorials: Reporting Services

SQL Server 2012 Books Online

Quick Step-by-Step

Microsoft

SQL Server 2012 Tutorials: Reporting Services

SQL Server 2012 Books Online

Summary: This book contains tutorials for SQL Server 2012 Reporting Services: Create a Data-Driven Subscription, Create a Basic Table Report, Create a Sample Report in Power View, and Optimize a Sample PowerPivot Model for Power View Reports.

Category: Quick Step-by-Step
Applies to: SQL Server 2012
Source: SQL Server Books Online (link to source content)
E-book publication date: June 2012

The example companies, organizations, products, domain names, email addresses, logos, people, places, and events depicted herein are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

This book expresses the author's views and opinions. The information contained in this book is provided without any express, statutory, or implied warranties. Neither the authors, Microsoft Corporation, nor its resellers, or distributors will be held liable for any damages caused or alleged to be caused either directly or indirectly by this book.

Contents

Reporting Services Tutorials (SSRS)

This section contains tutorials for SQL Server 2012 Reporting Services.

Create a Data-Driven Subscription (SSRS Tutorial)

Reporting Services provides data-driven subscriptions so that you can customize the distribution of a report based on dynamic list of subscribers that will receive the report. Data-driven subscriptions are typically created and maintained by report server administrators. The ability to create data-driven subscriptions requires expertise in building queries, knowledge of data sources that contain subscriber data, and elevated permissions on a report server.

Creating a Basic Table Report (SSRS Tutorial)

Follow the steps in this tutorial to learn how to create your first report. This tutorial shows you how to work with SQL Server Data Tools (SSDT) to create a data connection (data source), define a simple query (dataset), and configure a data region to contain your data.

See Also

Adventure Works for SQL Server 2012

Reporting Services Samples on CodePlex (http://go.microsoft.com/fwlink/?LinkId=198283)

TechNet WIKI: SQL Server 2012 Samples (http://go.microsoft.com/fwlink/?linkID=220734)

Picnic Dataset (Sample Data) (http://go.microsoft.com/fwlink/?linkID=219108)

Create a Basic Table Report (SSRS Tutorial)

This tutorial is designed to help you create a basic table report based on the database using Report Designer. You can also use Report Builder or the Report Wizard to create reports. In this tutorial, you will create a report project, set up connection information, define a query, add a Table data region, group and total some fields, and preview the report.

Note

To complete this tutorial, you must be running Reporting Services in native mode. If you are running Reporting Services in SharePoint integrated mode, the steps that use report server URLs do not work. For more information about Reporting Services modes, see Planning a Deployment Mode (Reporting

Services). For more information about deploying projects, see Deploying Reports, Models, and Shared Data Sources to a SharePoint Site.

Requirements

Your system must have the following installed to use this tutorial:

- Microsoft SQL Server 2012 database engine.
- The database. For more information, see Adventure Works for SQL Server 2012 (Adventure Works for SQL Server 2012) (http://go.microsoft.com/fwlink/?LinkId=245471).. For more information about support for SQL Server sample databases and sample code for SQL Server Express, see Databases and Samples Overview on the CodePlex Web site.
- SQL Server 2012 Reporting Services (SSRS).
- SQL Server SQL Server Data Tools (SSDT).

Note

SQL Server Data Tools is not supported on Itanium-based computers. However, support for SQL Server Data Tools is available for x64-based computers. If the SQL Server sample databases have been deployed on an Itanium-based computer, use BI Development Studio on either an x86-based or x64-based computer to modify and run the samples.

You must also have read-only permissions to retrieve data from the database.

Tasks

Lesson 1: Creating a Report Server Project

Lesson 2: Setting Up Connection Information (Reporting Services)

Lesson 3: Defining a Query for the Report (Reporting Services)

Lesson 4: Adding a Table to the Report

Lesson 5: Formatting a Report (Reporting Services)

Lesson 6: Adding Grouping and Totals (Reporting Services)

Note

When reviewing tutorials, we recommend that you add **Next** and **Previous** buttons to the document viewer toolbar. For more information, see Adding Next and Previous Buttons to Help.

See Also

Designing and Implementing Reports: Tutorials

Report Designer

Report Builder

Creating a Report Using Report Wizard

Reporting Services Samples

Considerations for Installing SQL Server Samples and Sample Databases

Lesson 1: Creating a Report Server Project (Reporting Services)

To create a report in SQL Server, you must first create a report server project where you will save your report definition (.rdl) file and any other resource files that you need for your report. Then you will create the actual report definition file, define a data source for your report, define a dataset, and define the report layout. When you run the report, the actual data is retrieved and combined with the layout, and then rendered on your screen, from where you can export it, print it, or save it.

In this lesson, you will learn how to create a report server project in SQL Server Data Tools (SSDT). A report server project is used to create reports that run on a report server.

Procedures

To create a report server project

1. Click **Start**, point to **All Programs**, point to , and then click **SQL Server Data Tools**. If this is the first time you have opened SQL Server Data Tools, click **Business Intelligence Settings** for the default environment settings.
2. On the **File** menu, point to **New**, and then click **Project**.
3. In the **Installed Templates** list, click **Business Intelligence**.
4. Click **Report Server Project**.
5. In **Name**, type **Tutorial**.
6. Click **OK** to create the project.

   The Tutorial project is displayed in Solution Explorer.

To create a new report definition file

1. In Solution Explorer, right-click **Reports**, point to **Add**, and click **New Item**.

   Note
   If the **Solution Explorer** window is not visible, from the **View** menu, click **Solution Explorer**.
2. In the **Add New Item** dialog box, under **Templates**, click **Report**.
3. In **Name**, type **Sales Orders.rdl** and then click **Add**.

   Report Designer opens and displays the new .rdl file in Design view.

Report Designer is a Reporting Services component that runs in SQL Server Data Tools (SSDT). It has two views: **Design** and **Preview**. Click each tab to change views.

You define your data in the **Report Data** pane. You define your report layout in **Design** view. You can run the report and see what it looks like in **Preview** view.

Next Task

You have successfully created a report project called "Tutorial" and added a report definition (.rdl) file to the report project. Next, you will specify a data source to use for the report. See Lesson 2: Setting Up Connection Information (Reporting Services).

See Also

Tutorial: Creating a Basic Table Report (Reporting Services)

Lesson 2: Specifying Connection Information (Reporting Services)

After you add a report to the Tutorial project, you need to define a *data source*, which is connection information the report uses to access data from either a relational database, multidimensional database, or other resource.

In this lesson, you will use the sample database as your data source. This tutorial assumes that this database is located in a default instance of SQL Server Database Engine that is installed on your local computer.

Procedures

To set up a connection

1. In the **Report Data** pane, click **New** and then click **Data Source...**.

   Note
   If the **Report Data** pane is not visible, from the **View** menu, click **Report Data**.

2. In **Name**, type .
3. Make sure **Embedded connection** is selected.
4. In **Type**, select **Microsoft SQL Server**.
5. In **Connection string**, type the following:

```
Data source=localhost; initial catalog=AdventureWorks2012
```

   This connection string assumes that SQL Server Data Tools (SSDT), the report server, and the database are all installed on the local computer and that you have permission to log on to the database.

   Note

If you are using SQL Server Express with Advanced Services or a named instance, the connection string must include instance information:

```
Data source=localhost\SQLEXPRESS; initial catalog=AdventureWorks2012
```

For more information about connection strings, see Connecting to a Data Source and Data Source Dialog Box, General Tab (Report Designer).

6. Click **Credentials** in the left pane and click **Use Windows Authentication (integrated security)**.
7. Click . data source AdventureWorks2012 is added to the **Report Data** pane.

Next Task

You have successfully defined a connection to the sample database. Next, you will create the report. See Lesson 3: Defining a Query for the Table Report (Reporting Services).

See Also

Connecting to a Data Source

Lesson 3: Defining a Dataset for the Table Report (Reporting Services)

After you define the data source, you need to define a dataset. In Reporting Services, data that you use in reports is contained in a *dataset*. A dataset includes a pointer to a data source and a query to be used by the report, as well as calculated fields and variables.

You can use the query designer in Report Designer to design the query. For this tutorial, you will create a query that retrieves sales order information from the **2008** database.

Procedures

To define a Transact-SQL query for report data

1. In the **Report Data** pane, click **New**, and then click **Dataset...**. The **Dataset Properties** dialog box opens.
2. In the **Name** box, type **AdventureWorksDataset**.
3. Click **Use a dataset embedded in my report**.
4. Make sure the name of your data source, AdventureWorks2012, is in the **Data source** text box, and that the **Query type** is **Text**.
5. Type, or copy and paste, the following Transact-SQL query into the **Query** box.

```
SELECT
    soh.OrderDate AS [Date],
    soh.SalesOrderNumber AS [Order],
    pps.Name AS Subcat, pp.Name as Product,
    SUM(sd.OrderQty) AS Qty,
    SUM(sd.LineTotal) AS LineTotal
FROM Sales.SalesPerson sp
    INNER JOIN Sales.SalesOrderHeader AS soh
       ON sp.BusinessEntityID = soh.SalesPersonID
    INNER JOIN Sales.SalesOrderDetail AS sd
       ON sd.SalesOrderID = soh.SalesOrderID
    INNER JOIN Production.Product AS pp
       ON sd.ProductID = pp.ProductID
    INNER JOIN Production.ProductSubcategory AS pps
       ON pp.ProductSubcategoryID = pps.ProductSubcategoryID
    INNER JOIN Production.ProductCategory AS ppc
       ON ppc.ProductCategoryID = pps.ProductCategoryID
GROUP BY ppc.Name, soh.OrderDate, soh.SalesOrderNumber,
pps.Name, pp.Name,
    soh.SalesPersonID
HAVING ppc.Name = 'Clothing'
```

6. (Optional) Click the **Query Designer** button. The query is displayed in the text-based query designer. You can toggle to the graphical query designer by clicking **Edit As Text**. View the results of the query by clicking the run **(!)** button on the query designer toolbar.

   You see the data from six fields from four different tables in the database. The query makes use of Transact-SQL functionality such as aliases. For example, the SalesOrderHeader table is called soh.

   Click **OK** to exit the query designer.

7. Click **OK** to exit the **Dataset Properties** dialog box.

   Your **AdventureWorksDataset** dataset and fields appear in the Report Data pane.

Next Task

You have successfully specified a query that retrieves data for your report. Next, you will create the report layout. See Lesson 4: Adding a Table to the Report (Reporting Services).

See Also

Query Design Tools in Reporting Services

SQL Server Connection Type (SSRS)

Tutorial: Writing Transact-SQL Statements

Lesson 4: Adding a Table to the Report (Reporting Services)

After the dataset is defined, you can start designing the report. You create a report layout by dragging and dropping data regions, text boxes, images, and other items that you want to include in your report to the design surface.

Items that contain repeated rows of data from underlying datasets are called *data regions*. A basic report will have only one data region, but you can add more, for example, if you want to add a chart to your tabular report. After you add a data region, you can add fields to the data region.

Procedures

To add a Table data region and fields to a report layout

1. In the **Toolbox**, click **Table**, and then click on the design surface and drag the mouse. Report Designer draws a table data region with three columns in the center of the design surface.

   Note
   The **Toolbox** may appear as a tab on the left side of the **Report Data** pane. To open the **Toolbox**, move the pointer over the **Toolbox** tab. If the **Toolbox** is not visible, from the **View** menu, click **Toolbox**.

2. In the **Report Data** pane, expand the **AdventureWorksDataset** dataset to display the fields.
3. Drag the Date field from the **Report Data** pane to the first column in the table.

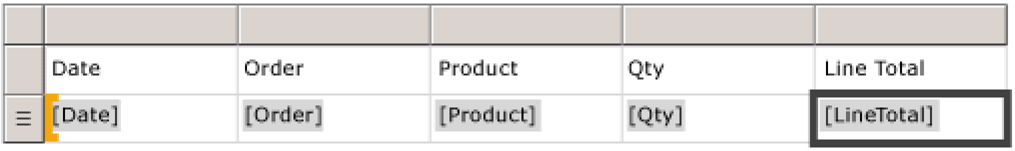
   When you drop the field into the first column, two things happen. First, the data cell will display the field name, known as the *field expression*, in brackets: `[Date]`. Second, a column header value is automatically added to Header row, just above the field expression. By default, the column is the name of the field. You can select the Header row text and type a new name.

4. Drag the Order field from the **Report Data** pane to the second column in the table.
5. Drag the Product field from the **Report Data** pane to the third column in the table.
6. Drag the Qty field to the right edge of the third column until you get a vertical cursor and the mouse pointer has a plus sign [+]. When you release the mouse button, a fourth column is created for `[Qty]`.
7. Add the LineTotal field in the same way, creating a fifth column.

Note

The column header is Line Total. Report Designer automatically creates a friendly name for the column by splitting LineTotal into two words.

The following diagram shows a table data region that has been populated with these fields: Date, Order, Product, Qty, and Line Total.



Date	Order	Product	Qty	Line Total
[Date]	[Order]	[Product]	[Qty]	[LineTotal]

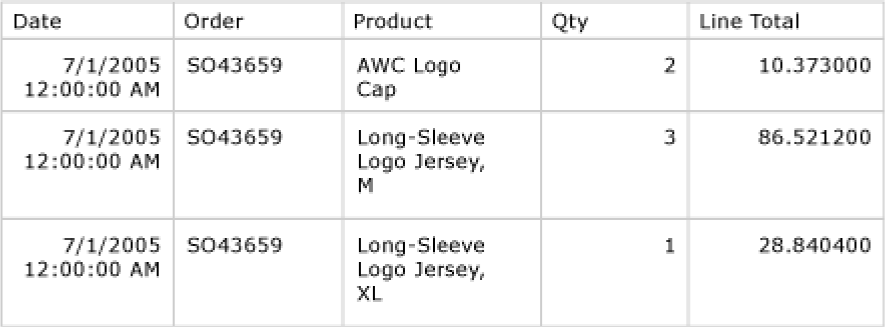
Preview Your Report

Previewing a report enables you to view the rendered report without having to first publish it to a report server. You will probably want to preview your report frequently during design time. Previewing the report will also run validation on the design and data connections so you can correct errors and issues before publishing the report to a report server.

To preview a report

- Click the **Preview** tab. Report Designer runs the report and displays it in Preview view.

  The following diagram shows part of the report in Preview view.



Date	Order	Product	Qty	Line Total
7/1/2005 12:00:00 AM	SO43659	AWC Logo Cap	2	10.373000
7/1/2005 12:00:00 AM	SO43659	Long-Sleeve Logo Jersey, M	3	86.521200
7/1/2005 12:00:00 AM	SO43659	Long-Sleeve Logo Jersey, XL	1	28.840400

Notice that the currency (in the Line Total column) has six places after the decimal, and the date has includes a time stamp. You're going to fix that formatting in the next lesson.

Note

On the **File** menu, click **Save All** to save the report.

Next Steps

You have successfully added a Table data region to your report, added fields to the data region, and previewed your report. Next, you will format column headers and date and currency values. See Lesson 5: Formatting a Report (Reporting Services).

See Also

Working with Table Data Regions

Adding Fields to a Report

Lesson 5: Formatting a Report (Reporting Services)

Now that you've added a data region and some fields to the Sales Orders report, you can format the date and currency fields and the column headers.

In this topic:

Format the Date

Format the Currency

Change Text Style and Column Widths

Format the Date

The Date field displays date and time information by default. You can format it to display only the date.

To format a date field

1. Click the **Design** tab.
2. Right-click the cell with the `[Date]` field expression and then click **Text Box Properties**.
3. Click **Number**, and then in the **Category** field, select **Date**.
4. In the **Type** box, select **January 31, 2000**.
5. Click .
6. Preview the report to see the change to the `[Date]` field and then change back to design view.

Top

Format the Currency

The LineTotal field displays a general number. Format it to display the number as currency.

To format a currency field

1. Right-click the cell with the `[LineTotal]` field expression and then click **Text Box Properties**.

2. Click **Number**, and in the **Category** field, select **Currency**.
3. If your regional setting is English (United States), the defaults should be:
   - **Decimal places: 2**
   - **Negative numbers: ($12345.00)**
   - **Symbol: $ English (United States)**

4. Select **Use 1000 separator (,)**.

   If the sample text is:**$12,345.00**, then your settings are correct.
5. Click .
6. Preview the report to see the change to the `[LineTotal]` field and then change back to design view.

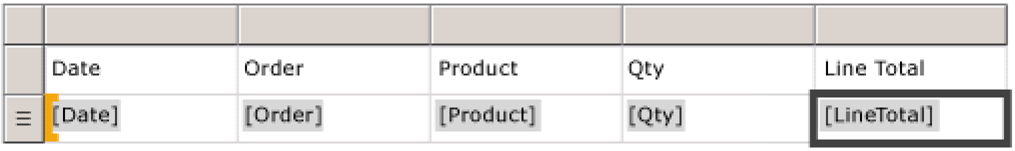
Top

Change Text Style and Column Widths

You can also change the formatting of the header row to differentiate it from the rows of data in the report. Lastly, you will adjust the widths of the columns.

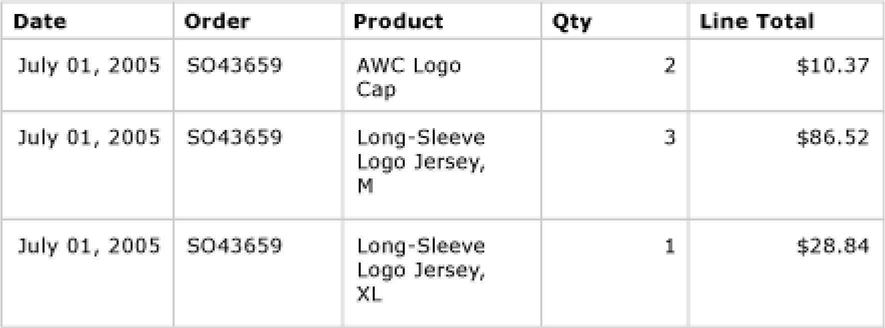
To format header rows and table columns

1. Click the table so that column and row handles appear above and next to the table.



Date	Order	Product	Qty	Line Total
[Date]	[Order]	[Product]	[Qty]	[LineTotal]

   The gray bars along the top and side of the table are the column and row handles.
2. Point to the line between column handles so that the cursor changes into a double arrow. Drag the columns to the size you want.
3. Select the row containing column header labels and from the **Format** menu, point to **Font** and then click **Bold**.
4. To preview your report, click the **Preview** tab. It should look something like this:



Date	Order	Product	Qty	Line Total
July 01, 2005	SO43659	AWC Logo Cap	2	$10.37
July 01, 2005	SO43659	Long-Sleeve Logo Jersey, M	3	$86.52
July 01, 2005	SO43659	Long-Sleeve Logo Jersey, XL	1	$28.84

5. On the **File** menu, click **Save All** to save the report.

Top

Next Steps

You have successfully formatted column headers and date and currency values. Next, you will add grouping and totals to your report. See Lesson 6: Adding Grouping and Totals (Reporting Services).

See Also

Formatting Numeric and Date Values in a Report

Understanding Rendering Behaviors

Lesson 6: Adding Grouping and Totals (Reporting Services)

Add grouping and totals to your report to organize and summarize your data.

In this topic:

To group data in a report

To add totals to a report

To add a daily total to a report

To add a grand total to a report

To Publish the Report to the Report Server (Optional)

Procedures

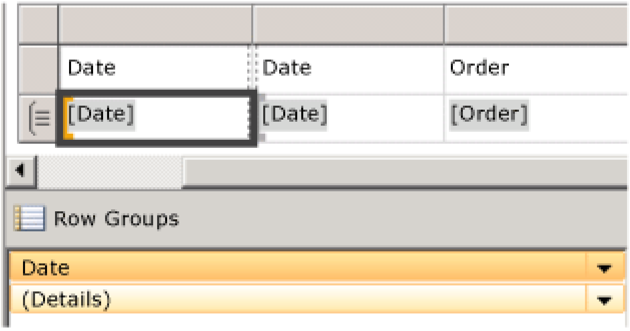
To group data in a report

1. Click the **Design** tab.
2. If you do not see the **Row Groups** pane , right-click the design surface and click

view and then click **Grouping**.

3. From the **Report Data** pane, drag the **Date** field to the **Row Groups** pane. Place it above the row called **(Details)**.

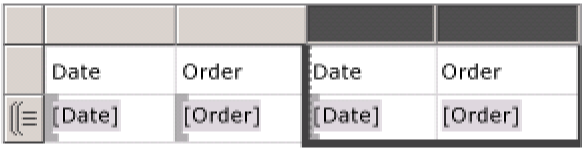
   Note that the row handle now has a bracket in it, to show a group. The table now also has two Date columns -- one on either side of a vertical dotted line.

4. From the **Report Data** pane, drag the **Order** field to the **Row Groups** pane. Place it below Date and above **(Details)**.

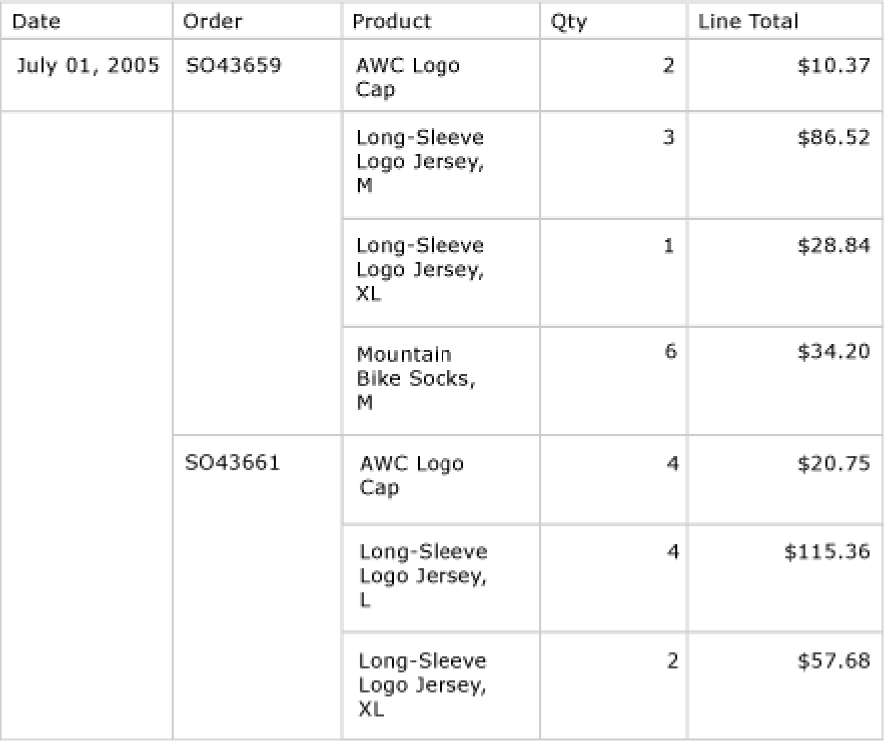
   Note that the row handle now has two brackets in it, to show two groups. The table now has two **Order** columns, too.

5. Delete the original Date and Order columns to the **right** of the double line. This removes this individual record values so that only the group value is displayed. Select the column handles for the two columns, right-click and click **Delete Columns**.

   You can format the column headers and date again.

6. Switch to the **Preview** tab to preview the report. It should look similar to the following illustration:



Date	Order	Product	Qty	Line Total
July 01, 2005	SO43659	AWC Logo Cap	2	$10.37
		Long-Sleeve Logo Jersey, M	3	$86.52
		Long-Sleeve Logo Jersey, XL	1	$28.84
		Mountain Bike Socks, M	6	$34.20
	SO43661	AWC Logo Cap	4	$20.75
		Long-Sleeve Logo Jersey, L	4	$115.36
		Long-Sleeve Logo Jersey, XL	2	$57.68

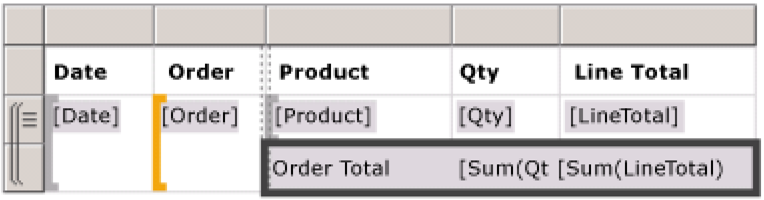
Top

To add totals to a report

1. Switch to Design view.
2. Right-click the data region cell that contains the field `[LineTotal]`, and click **Add Total**.

   This adds a row with a sum of the dollar amount for each order.
3. Right-click the cell that contains the field `[Qty]`, and click **Add Total**.

   This adds a sum of the quantity for each order to the totals row.
4. In the empty cell to the left of `Sum[Qty]`, type the label "**Order Total"**.
5. You can add a background color to the totals row. Select the two sum cells and the label cell.
6. On the **Format** menu, click **Background Color**, click **Light Gray**, and click **OK**.



Date	Order	Product	Qty	Line Total
[Date]	[Order]	[Product]	[Qty]	[LineTotal]
		Order Total	[Sum(Qt	[Sum(LineTotal)

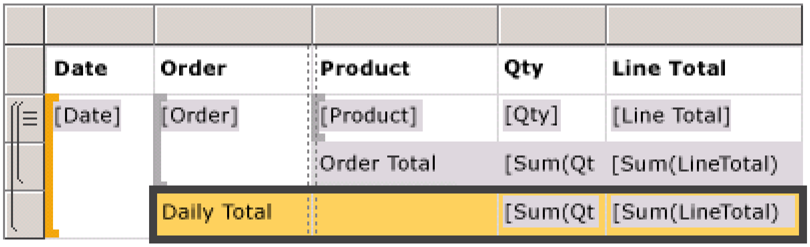
Top

To add a daily total to a report

1. Right-click the Order cell, point to **Add Total**, and click **After**.

   This adds a new row containing sums of the quantity and dollar amount for each day, and the label "**Total**" in the Order column.

2. Type the word **Daily** before the word **Total** in the same cell, so it reads **Daily Total**.
3. Select the **Daily Total** cell, the two **Sum** cells and the empty cell between them.
4. On the **Format** menu, click **Background Color**, click **Orange**, and click **OK**.



Date	Order	Product	Qty	Line Total
[Date]	[Order]	[Product]	[Qty]	[Line Total]
		Order Total	[Sum(Qt	[Sum(LineTotal)
	Daily Total		[Sum(Qt	[Sum(LineTotal)

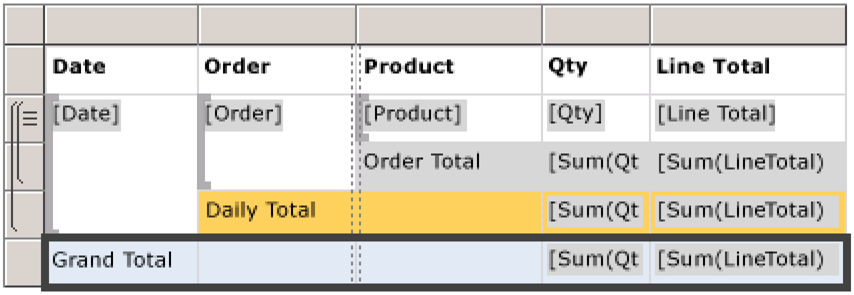
Top

To add a grand total to a report

1. Right-click the Date cell, point to **Add Total**, and click **After**.

   This adds a new row containing sums of the quantity and dollar amount for the entire report, and the **Total** label in the **Date** column.

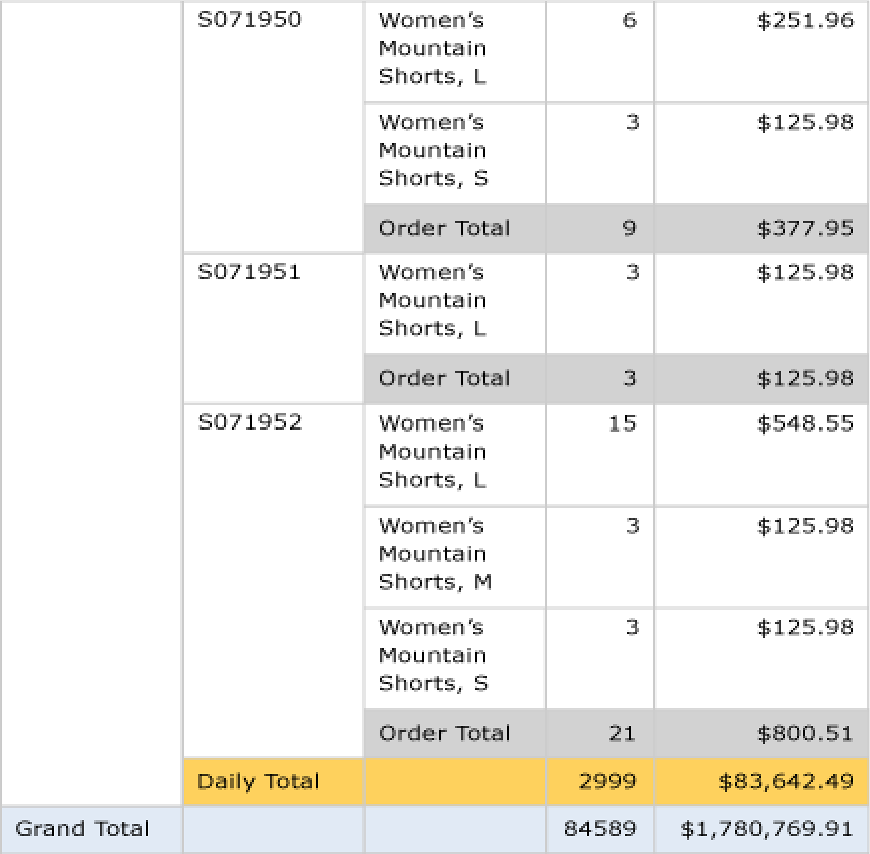
2. Type the word **Grand** before the word **Total** in the same cell, so it reads **Grand Total**.
3. Select the **Grand Total** cell, the two **Sum** cells and the empty cells between them.
4. On the **Format** menu, click **Background Color**, click **Light Blue**, and click **OK**.



Date	Order	Product	Qty	Line Total
[Date]	[Order]	[Product]	[Qty]	[Line Total]
		Order Total	[Sum(Qt	[Sum(LineTotal)
	Daily Total		[Sum(Qt	[Sum(LineTotal)
Grand Total			[Sum(Qt	[Sum(LineTotal)

5. Click Preview.

   The last page should look something like this:



	S071950	Women’s Mountain Shorts, L	6	$251.96
		Women’s Mountain Shorts, S	3	$125.98
		Order Total	9	$377.95
	S071951	Women’s Mountain Shorts, L	3	$125.98
		Order Total	3	$125.98
	S071952	Women’s Mountain Shorts, L	15	$548.55
		Women’s Mountain Shorts, M	3	$125.98
		Women’s Mountain Shorts, S	3	$125.98
		Order Total	21	$800.51
	Daily Total		2999	$83,642.49
Grand Total			84589	$1,780,769.91

[Top](#)

To Publish the Report to the Report Server (Optional)

1. An optional step is to publish the completed report to the native mode report server so you can view the report from Report Manager.
2. On the toolbar click **Project** and then click **tutorial Properties...**
3. In the **TargetServerURL** type the name of the name of your report server, for example **http://<servername>/reportserver**
4. Click **OK**
5. On the toolbar click **Build** and then click **Deploy tutorial**.

   If you see a message similar to the following in the output window, it indicates a successful deployment.

   If you see an error message similar to the following, verify you have permissions on the report server and you have started SQL Server Data Tools with administrator privileges.

6. Start Report Manager with administrator privileges, for example, right-click the icon for Internet Explorer and click **Run as administrator**.

   Browse to the Report Manager URL, for example: `http://<server name>/reports`.

7. Browse to the folder that contains the report and click the name of the report **Sales Orders** to view the rendered report in the browser.

Top

Next Steps

You have successfully completed the Creating a Basic Table Report tutorial.

See Also

Filtering, Grouping, and Sorting Data (Report Builder 3.0 and SSRS)

Create a Data-Driven Subscription (SSRS Tutorial)

Reporting Services provides data-driven subscriptions so that you can customize the distribution of a report based on dynamic subscriber data. Data-driven subscriptions are intended for the following kinds of scenarios:

- Distributing reports to a large recipient pool whose membership may change from one distribution to the next. For example, distribute a monthly report to all current customers.
- Distributing reports to a specific group of recipients based on predefined criteria. For example, send a sales performance report to the top ten sales managers in an organization.

What You Will Learn

This tutorial shows you how to use data-driven subscriptions using a simple example to illustrate the concepts.

This tutorial is divided into three lessons:

Lesson 1: Creating a Sample Subscriber Database

In this lesson you will learn how to create a local SQL Server database that contains subscriber information.

Lesson 2: Modify the Report Data Source Properties

In this lesson, you will learn how to modify report data source properties so that the report can run unattended. Unattended processing requires stored credentials. You will also modify the report dataset to include a parameter that is supplied by the subscriber data.

Lesson 3: Define a Data-Driven Subscription

In this lesson you will learn how to define a data-driven subscription. This lesson guides you through each page in the Data-Driven Subscription Wizard.

Requirements

Data-driven subscriptions are typically created and maintained by report server administrators. The ability to create data-driven subscriptions requires expertise in building queries, knowledge of data sources that contain subscriber data, and elevated permissions on a report server.

The tutorial will use the report created in the tutorial Creating a Basic Table Report (SSRS Tutorial) and data from

Your system must have the following installed to use this tutorial:

- An edition of SQL Server that supports data-driven subscriptions. For more information, see Editions and Components of SQL Server 2012.
- The report server must be running in native mode. The user interface described in this tutorial is based on a native mode report server. Subscriptions are supported on SharePoint mode report servers but the user interface will be different than what is described in this tutorial.
- SQL Server Agent service must be running.
- A report that includes parameters. This tutorial assumes the sample report, `Sales Orders` you create using the tutorial Creating a Basic Table Report (SSRS Tutorial).
- The sample database, which provides data to the sample report.
- A role assignment that includes the Manage all subscriptions task on the sample report. This task is required for defining a data-driven subscription. If you are an administrator on the computer, the default role assignment for local administrators provides the permissions necessary for creating data-driven subscriptions. For more information, see Using Role-Based Security.
- A shared folder for which you have write permissions. The shared folder must be accessible over a network connection.

Estimated time to complete the tutorial: 30 minutes. An additional 30 minutes if you have not completed the basic report tutorial.

See Also

Data-Driven Subscriptions

Creating a Basic Table Report (SSRS Tutorial)

Lesson 1: Creating a Sample Subscriber Database

Before you can define a data-driven subscription, you must have a data source that provides subscription data. In this step, you will create a small database to store the subscription data used in this tutorial. Later, when the subscription is processed, the report server retrieves this data and uses it to customize report output, delivery options, and report presentation format.

This lesson assumes you are using Management Studio to create a SQL Server 2012 database.

Procedures

To create a sample Subscriber database

1. Start Management Studio, and open a connection to a Database Engine.
2. Right-click on Databases, select **New Database...**.
3. In the New Database dialog box, in Database Name, type *Subscribers*. Click .
4. Click the **New Query** button on the toolbar.
5. Copy the following Transact-SQL statements into the empty query:

```
Use Subscribers
CREATE TABLE [dbo].[OrderInfo] (
    [SubscriptionID] [int] NOT NULL PRIMARY KEY ,
    [Order] [nvarchar] (20) NOT NULL,
    [FileType] [bit],
    [Format] [nvarchar] (20) NOT NULL ,
) ON [PRIMARY]
GO

INSERT INTO [dbo].[OrderInfo] (SubscriptionID, [Order],
FileType, Format)
VALUES ('1', 'so43659', '1', 'IMAGE')
INSERT INTO [dbo].[OrderInfo] (SubscriptionID, [Order],
FileType, Format)
VALUES ('2', 'so43664', '1', 'MHTML')
INSERT INTO [dbo].[OrderInfo] (SubscriptionID, [Order],
FileType, Format)
VALUES ('3', 'so43668', '1', 'PDF')
INSERT INTO [dbo].[OrderInfo] (SubscriptionID, [Order],
FileType, Format)
VALUES ('4', 'so71949', '1', 'Excel')
GO
```

6. Click **! Execute** on the toolbar.
7. Use a SELECT statement to verify that you have three rows of data. For example:
   `select * from OrderInfo`

Next Steps

You have successfully created the subscription data that will drive report distribution and vary the report output for each subscriber. Next, you will modify the data source properties of the report you will distribute to subscribers. Modifying the data source properties is done to prepare the report for data-driven subscription delivery. You will also modify the report design to include a parameter that the subscription will use with the subscriber data. Lesson 2: Modifying Report Data Source Properties.

See Also

Create a Data-Driven Subscription (SSRS Tutorial)

How to: Create a database (SQL Server Management Studio)

Adding Rows with INSERT

Creating a Basic Table Report (SSRS Tutorial)

Lesson 2: Modifying the Report Data Source Properties

In this lesson, you will use Report Manager to select a report that will be delivered to recipients. The data-driven subscription that you will define will distribute the **Sales Order** report created in the tutorial Creating a Basic Table Report (SSRS Tutorial). In the steps that follow, you will modify the data source connection information used by the report to get data. Only reports that use **stored credentials** to access a report data source can be distributed through a data-driven subscription. Stored credentials are necessary for unattended report processing.

You will also modify the dataset and report to use a parameter to filter the report on the `[Order]` so the subscription can output different instances of the report for specific orders and rendering formats.

In this topic:

To Modify the Data Source Properties

To Modify the AdventureWorksDataset

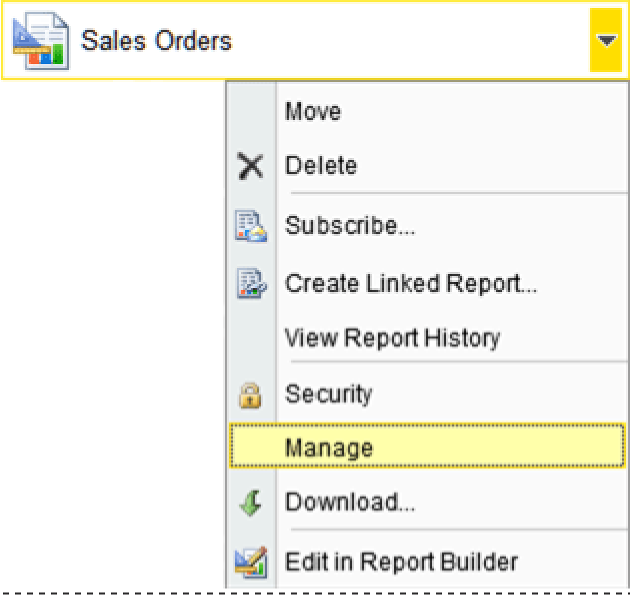
To Add a Report Parameter and Re-publish the Report

To Re-deploy the Report

Procedures

To Modify the Data Source Properties

1. Start Report Manager with administrator privileges, for example, right-click the icon for Internet Explorer and click **Run as administrator**.
2. Browse to the folder containing the **Sales Orders** report and in the context menu of the report, click **Manage**.



3. Click the **Data Sources** tab.
4. For **Connection Type**, select **Microsoft SQL Server**.
5. The custom data source connection string will be the following and it assumes that the sample database is on a local database server:

```
Data source=localhost; initial catalog=AdventureWorks2012
```

6. Click **Credentials stored securely in the report server**.
7. Type your user name (use the format *domain\user*) and password. If you do not have permission to access the database, specify a login that does.
8. Click **Use as windows credentials when connecting to the data source**, and then click **OK**. If you are not using a domain account (for example, if you are using a SQL Server login), do not click this checkbox.
9. Click **Test Connection** to verify you can connect to the data source.
10. Click **Apply**.
11. View the report to verify that the report runs with the credentials you specified. To view the report, click the **View** tab. Note that once the report is open, you

must select an Employee name and then click the **View Report** button to view the report.

⬆ Top

▶To Modify the AdventureWorksDataset

1. Open the Sales Orders report in SQL Server Data Tools
2. Right-click the dataset `AdventureWorksDataset` and click **Dataset Properties**.
3. Add the statement `WHERE (UPPER(SalesOrderNumber) =UPPER(@OrderNumber) or @OrderNumber IS NULL)` before the `Group By` statement. The full query syntax is the following:

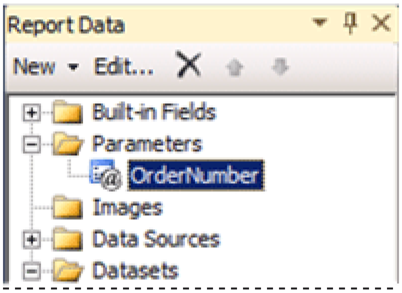
```
SELECT soh.OrderDate AS Date, soh.SalesOrderNumber AS
[Order], pps.Name AS Subcat, pp.Name AS Product,
SUM(sd.OrderQty) AS Qty, SUM(sd.LineTotal)  AS LineTotal
FROM Sales.SalesPerson AS sp INNER JOIN
  Sales.SalesOrderHeader AS soh ON sp.BusinessEntityID =
soh.SalesPersonID INNER JOIN
   Sales.SalesOrderDetail AS sd ON sd.SalesOrderID =
soh.SalesOrderID INNER JOIN
   Production.Product AS pp ON sd.ProductID = pp.ProductID
INNER JOIN
   Production.ProductSubcategory AS pps ON
pp.ProductSubcategoryID = pps.ProductSubcategoryID
INNER JOIN
    Production.ProductCategory AS ppc ON
ppc.ProductCategoryID = pps.ProductCategoryID

WHERE (UPPER(SalesOrderNumber) =UPPER(@OrderNumber) or
@OrderNumber IS NULL)

GROUP BY ppc.Name, soh.OrderDate, soh.SalesOrderNumber,
pps.Name, pp.Name, soh.SalesPersonID
HAVING (ppc.Name = 'Clothing')
```

4. Click **OK**

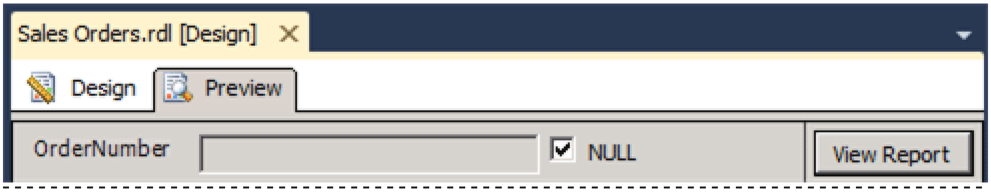
⬆ Top

▶To Add a Report Parameter and Republish the Report

1. In the **Report Data** pane click **New** and then click **Parameter…**
2. In **Name**, type `OrderNumber`.
3. In **Prompt**, type `OrderNumber`.
4. Select **Allow blank value ("")**.
5. Select **Allow null value**.
6. Click **OK**. The parameter will be added to the **Report Data pane** and it will look like the following image:



7. Click the **Preview** tab to run the report.Note the parameter input box at the top of the report. You can either:
   - Click View Report to see the full report without using a parameter.
   - Unselect the Null option and type an order number, for example so71949 to view only the one order in the report.



8. Re-deploy the report so the subscription configuration in the next lesson can utilize the changes you made in this lesson. For more information on the project properties used in the table tutorial, see section 'To Publish the Report to the Report Server (Optional)' of Lesson 6: Adding Grouping and Totals (Reporting Services).

Top

To Re-deploy the Report

1. Re-deploy the report so the subscription configuration in the next lesson can utilize the changes you made in this lesson. For more information on the project properties used in the table tutorial, see section 'To Publish the Report to the

Report Server (Optional)' of Lesson 6: Adding Grouping and Totals (Reporting Services).

2. On the toolbar click **Build** and then click **Deploy tutorial**.

Top

Next Steps

You successfully configured the report to get data using stored credentials. Next, you specify the subscription using the Data-Driven Subscription pages in Report Manager. See Lesson 3: Define a Data-Driven Subscription.

See Also

Setting Data Source Properties in Reporting Services

Specifying Credential and Connection Information

Create a Data-Driven Subscription (SSRS Tutorial)

Creating a Basic Table Report (SSRS Tutorial)

Lesson 3: Defining a Data-Driven Subscription

In this lesson, you use the data-driven subscription pages to connect to a subscription data source, build a query that retrieves subscription data, and map the result set to report and delivery options.

> **Note**
> Before you start, verify that SQL Server Agent service is running. If it is not running, you cannot save the subscription.

This lesson assumes you completed Lesson 1 and Lesson 2 and that the report data source uses stored credentials. For more information, see Lesson 2: Modify the Report Data Source Properties

In this topic:

Start the Data-Driven Subscription Wizard

Step 1 - Define a Description

Step 2 - Define a Connection to the Subscriber Data Source

Step 3 - Define a query to retrieve subscriber data

Step 4 - Set Delivery Options

Step 5 - Configure a Parameter Value to Very Report Output

Step 6 - To Schedule a Subscription

Procedures

Start the Data-Driven Subscription Wizard

1. In Report Manager, click **Home**, and navigate to the folder containing the **Sales Orders** report.
2. In the context menu of the report, click **Manage**, and then click the **Subscriptions** tab.
3. Click **New Data-driven Subscription**. If you do not see this button, you do not have Content Manager permissions.

Top

Step 1 - Define a description

1. Type **Sales Order delivery** in description.
2. Select **Windows FileShare** for **Specify how recipients are notified**.
3. Select **Specify for this subscription only**, and then click **Next**.

Top

Step 2 - Define a Connection to the Subscriber Data Source

1. Select **Microsoft SQL Server** as the data source type.
2. In Connection string, type the following connection string:

   ```
   data source=localhost; initial catalog=Subscribers
   ```

   Note
   Subscribers is the database you created in lesson 1.
3. Click **Credentials stored securely in the report server**.
4. In **User Name** and **Password**, type your domain user name and password. Include both the domain and user account when specifying **User Name**.

   Note
   Credentials used to connect to a subscriber data source are not passed back to Management Studio. If you modify the subscription later, you must retype the password used to connect to the data source.
5. Select **Use as windows credentials when connecting to the data source**, and then click **Next**.

Top

Step 3 - Define a Query to Retrieve Subscriber data

1. In the query box, type the following query:

   ```
   Select * from OrderInfo
   ```
2. Specify a time-out of 30 seconds.
3. Click **Validate**, and then click **Next**.

Top

Step 4 - Set Delivery Options

1. For **File name**, select **Get the value from the database**. Select the field **Order**.
2. For **Path**, select **Specify a static value**. In Setting Value, type the name of a public file share for which you have write permissions (for example, `\\mycomputer\public\myreports`).
3. For **Render Format**, select **Get the value from the database**. Select **Format**.
4. For **Write mode**, select **Specify a static value** and select **AutoIncrement**.
5. For **File Extension**, select **Specify a static value** and select **True**.
6. For **User name**, select **Specify a static value**. Type your domain user account. Enter it in this format: `<domain>\<account>`. The user account needs to have permissions to the path you configured in the previous steps.
7. For **Password**, select **Specify a static value**. Type your password. Be sure that you type the password carefully. The wizard does not validate the password.
8. Click **Next.**

Top

Step 5 - Configure a Parameter Value to Very Report Output

1. For **OrderNumber**, select **Get the value from the database**. In Value, select **Order**. Click **Next.**

Top

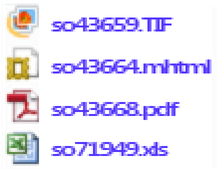
Step 6 - To Schedule a Subscription

1. Click **On a schedule created for this subscription**, and then click **Next**.
2. In **Schedule Details**, click **Once**.
3. Specify a start time that is a few minutes ahead of the current time.
4. Click **Finish**.

Top

Next Steps

When the subscription runs, four report files will be delivered to the file share you specified, one for each order in the *Subscribers* data source. Each delivery should be unique in terms of data (the data should be order-specific), rendering format, and file format. You can open each report from the shared folder to verify that each version is customized based on the subscription options you defined.



so43659.TIF

so43664.mhtml

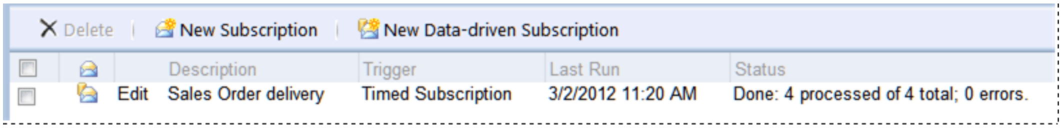
so43668.pdf

so71949.xls

The subscription page in Report Manager will contain the **Last Run** date and **Status** of the subscription.

Note

Refresh the page after the subscription runs to see the updated information.



☐			Description	Trigger	Last Run	Status
☐		Edit	Sales Order delivery	Timed Subscription	3/2/2012 11:20 AM	Done: 4 processed of 4 total; 0 errors.

This step concludes the tutorial "Defining a Data-Driven Subscription". To learn more about other Reporting Services tutorials, see Reporting ServicesTutorials (SSRS).

Top

See Also

Tutorial: Creating a Data-Driven Subscription

Distributing Reports Through Subscriptions

Data-Driven Subscriptions

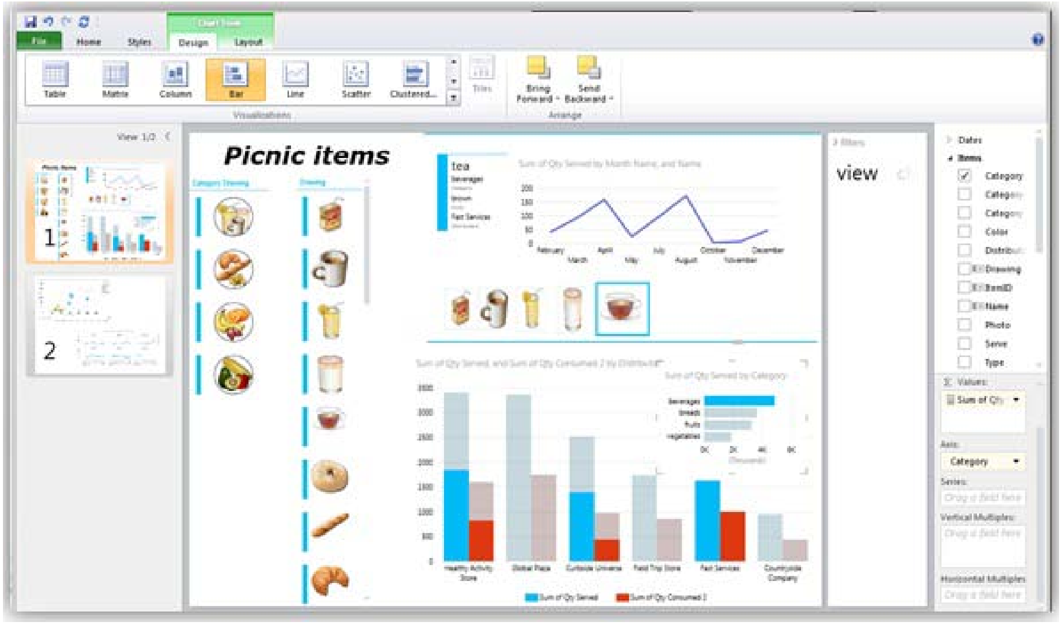
Creating, Modifying, and Deleting Data-Driven Subscriptions

Using an External Data Source for Subscriber Data

Tutorial: Create a Sample Report in Power View

In this tutorial, you create a Power View report that demonstrates what Power View can do, using a small sample model created in PowerPivot for Excel. You create tables and convert them into a number of different visualizations. You create *tiles* and add visualizations to them. You create multiple pages, or *views*, in your report. You create a bubble chart and add a play axis to it, so you can see how the values change over time. You create *small multiples* – a set of small charts, one per series, with the same axes. Then you experiment with different ways to filter the data in the visualizations. In the end, you export your report to PowerPoint.

A report similar to the one that you can make by following the steps in this tutorial



Prerequisites

To do this tutorial, you need to:

- Have a computer that meets the System requirements for Project Crescent.
- Download and install Power View samples, including the image files, from the Download Center. For more information, see Power View and PowerPivot HelloWorldPicnic Samples for SQL Server 2012 in SQL Server 2012 Samples .
- Install PowerPivot for Excel.

Note

HelloWorld_Picnic is a tiny sample data model that demonstrates how to set data model properties to best present that data in Power View reports. In the scenario, the data is used to plan and track food and drink for picnics. For more information, see Power View Samples for SQL Server 2012.

In This Article

Sort the values in a chart

Arrange items in the view

Highlighting and slicers

Cards

Tiles

Compare highlighting and slicer filtering

Add another view

Bubble charts

Add category group colors to a bubble chart

Small multiples

View and visualization filters

View your report

Save your report

Export your report to PowerPoint

Before you start the tutorial

Before you can start the tutorial, save the HelloWorldPicnicPowerViewRTM.xlsx file to a:

- **SharePoint document library**. If you save it to a SharePoint document library, create a shared data source that points to the HelloWorldPicnicPowerViewRTM.xlsx file that you saved to the SharePoint document library.
- **PowerPivot Gallery**. If you save it to a PowerPivot Gallery, it doesn't need the shared data source. You can open Power View directly from the gallery.

Save the images to the associated SharePoint document library.

To create a shared data source (RSDS) file to point to the PowerPivot (XLSX) file

1. In Shared Documents, create a HelloWorldPicnicSQL2012 folder, and save the HelloWorldPicnicPowerViewRTM.xlsx file there.
2. In the HelloWorldPicnicSQL2012 folder, click the **Documents** tab, click the arrow next to **New Document**, and then click **Report Data Source**.

   If there is no **Report Data Source** option on the **New Document** tab, then someone with adequate permissions on the SharePoint site needs to add that content type to the site. For more information, see Add Report Server Content Types to a Library (Reporting Services in SharePoint Integrated Mode).
3. In the **Name** box, type **HelloWorldPicnicRSDS**.

   Note the RSDS file extension.
4. In the **Data Source Type** box, click **Microsoft BI Semantic Model for Power View**.

5. The connection string for an XLSX file is the full URL to the file, including the file name. For example:

   Data Source='http://<myserver>/Shared%20Documents/HelloWorldPicnicSQL2012/HelloWorldPicnicPowerViewRTM.xlsx'

6. For **Credentials**, click **Windows authentication (integrated) or SharePoint user**.
7. Click **Test Connection**.

   You should see a message that the connection test was successful.

8. Click **OK**.

To save the images

1. Create an Images folder in the HelloWorldPicnicSQL2012 folder. For example:

   http://<myserver>/Shared Documents/HelloWorldPicnicSQL2012/Images

2. In the Images folder, save the images that you downloaded from the Download Center. For more information, see the Prerequisites section at the beginning of this tutorial.

To update the links to the images

1. Open HelloWorldPicnicPowerViewRTM.xlsx in Excel.
2. On the **PowerPivot** tab, click **PowerPivot window**.

   If you don't see a PowerPivot tab, you may need to install PowerPivot for Excel.

3. In the Items worksheet, find the URL in the Photo, Drawing, Category Drawing, and Category Photo columns.

   For example, the URL in the Photo column is: http://localhost/Shared Documents/HelloWorldPicnicSQL2012/HelloWorldPicnicImages/rs_crescent_picnic_image_.

4. In each column, update localhost in the URL with the name of your server.

Start the tutorial

1. If you saved HelloWorldPicnicPowerViewRTM.xlsx in a SharePoint Server 2010 document library, click the arrow next to the **HelloWorldPicnicRSDS**.rsds shared data source, and then click **Create Power View Report**.

   If you saved HelloWorldPicnicPowerViewRTM.xlsx to the PowerPivot Gallery, then click the **Create Power View Report** icon for .

   Power View opens with the blank view in the center, the ribbon across the top, and field list on the right. In the fields section of the field list are three tables. To begin with, the tables are collapsed, showing just their names:

- Dates
- Items
- Quantities

These tables make up the data model on which the Power View report is based.

2. To expand the tables, click the arrows next to the table names.

   Now you can see the fields in each table. These may include *calculated columns* (marked with a Sigma ∑ symbol), *measures* (marked with a small calculator symbol), and *row label fields* (marked with a gray and white icon).

For more information about calculated fields and measures, see Measure and Non-Measure Fields in Power View. For information about the Power View design environment in general, see Project Crescent Design Experience.

Set the title

To change the title text and formatting

1. Click in the default title box and type a title: **Picnic items**.
2. Select the text and click the **Italic** button, and then make the text **Segoe (Bold)**.
3. Drag the sizing handle on the right side of the title text box to make it half the width and in the top-left corner of the view.

Bar and line charts

To create bar and line charts

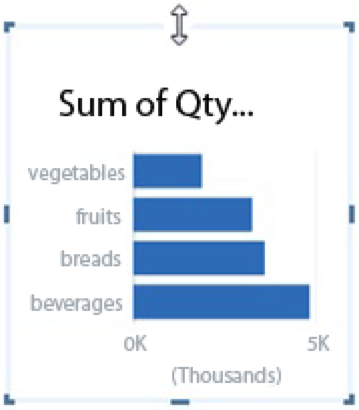
1. Click the blank view.
2. In the fields section of the field list, in the Items table, drag the Category field to the blank view.

   You now have a one-column table in the view.
3. In the fields section of the field list, in the Quantities table, drag Sum of Qty Served to the one-column Category table.

   The table now has two columns with a total.
4. In the **Visualizations** gallery, click the **Bar** chart.
5. Hover over the edge of the chart until the double-headed sizing arrows appear over one of the sizing handles. Drag the sizing handle until you can see all the bars of the chart.
6. Hover over the edge of the chart until the hand icon appears. The hand icon

means you can move the chart. Drag the chart to the lower-left corner of the report for now.

Bar chart with sizing handles



Copy and paste a visualization

▶To copy and paste

1. With the bar chart selected, on the **Home** menu click **Copy**.

   You may see a message asking if you want to allow this Web application to access your clipboard. If you do, click **Yes**.
2. Click off the bar chart on the blank canvas, and then click **Paste**.

   A second bar chart is pasted on the view.
3. Make sure the copy of the chart is selected. In the layout section of the field list in the **Axis** box, click the arrow next to Category and then click **Remove Field**.
4. In the fields section of the field list in the Items table, click the arrow next to the Distributors field and click **Add as Axis**.
5. In the **Visualizations** gallery, click the **Column** chart.

Add a second series to a chart

▶

1. In the fields section of the field list in the Quantities table, click the dropdown arrow next to the Sum of Qty Consumed field and click **Add to Values**.
2. The chart is now a stacked column chart, with the Qty Served and Qty Consumed values stacked on top of each other.

3. Click the dropdown arrow in the **Visualizations** gallery and click the **Clustered Column** chart.

   Now the Qty Served and Qty Consumed values are side by side.

4. Hover over the edge of the chart until the hand appears, and then drag the chart to the lower-right corner of the report.
5. Hover over the upper-right corner of the chart until the double-headed sizing arrows appear. Drag the sizing handle until the chart is half as tall as the view, and half as wide.

Sort the values in a chart

▶

1. When you hover over the Distributors column chart, you see **sort by Distributors asc**. Click the arrow next to **Distributors** and click **Qty Served**.

   Now the bars are sorted from smallest to largest quantity served.

2. Click **asc**. It changes to **desc**, and the bars are sorted from largest to smallest quantity served.

Move the chart legend

▶

- With the Distributors chart selected, click the **Layout** tab in the **Chart Tools** contextual tab, click **Legend**, and then click **Show Legend at Bottom**.

Arrange items in the view

▶

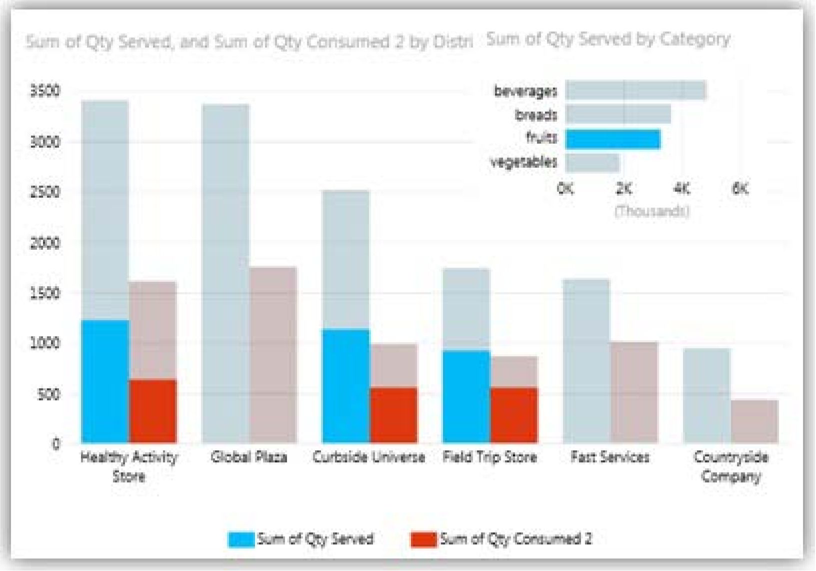
1. Drag the small Serve bar chart so that it is on top and in the upper-right corner of the large Distributors column chart.

   By default, Category, the first chart you created, will be ‘behind’ or ‘under’ the second chart, Distributors.

2. With the Category chart still selected, click **Bring Forward**.

   If you no longer have the Category chart selected and it’s now completely covered by the Distributors chart, just select the Distributors chart and click **Send Backward**.

The Distributors chart with the Categories chart inset on it, with fruits highlighted.



Highlighting and slicers

You can filter and highlight data in your report in a few different ways. All of them allow you to select one or multiple values.

▶ To highlight one chart with another chart

- In the Category bar chart, click the fruits bar.

  Notice that the values in the Distributors column chart are highlighted to show the amount of fruit from each distributor. Columns for distributors with no fruit are completely grayed.

  Tip
  To clear the selection, click in the Category chart, but not on any bar.

▶ To create a slicer

1. Click the blank part of the view, so none of the visualizations are selected.
2. In the Items table in the fields section of the field list, select Category Drawing.
3. On the **Design** tab, click **Slicer**.

   The slicer contains four filtering buttons. Each button features the image of one of the four categories.

   Note
   If you don't see the images of the categories, the images may be missing.

See the section about saving images in Before you start the tutorial.

4. Drag this slicer to the upper-left corner of the view, under the title.
5. Resize it so you can see all four categories.

To create a second slicer

1. Click the blank part of the view, so none of the visualizations are selected.
2. In the Items table in the fields section of the field list, select Drawing.
3. On the **Design** tab, click **Slicer**.

   This slicer contains filtering buttons, one for each item, with an image for each item.
4. Drag it to the right of the Category Drawing slicer and make it as tall as the view allows.
5. Click one of the categories in the Category Drawing slicer.

   Note the behavior is different from highlighting in the chart.

   - It filters everything in all the visualizations on the report, rather than just highlighting. For example, the Categories bar chart now just has one bar -- the category that you've selected in the slicer. The Distributors column chart just has the columns for the distributors in that category.
   - It filters the items in the other slicer. If you select beverages in the Category Drawing slicer, then the Drawing slicer will have only beverages.
6. To show all values again, click the **Clear filter** icon in the upper right corner of the slicer.

For more information, see Filtering, Highlighting, and Slicers in Power View.

Cards

To convert a table to a card

1. Click the blank part of the view to start a new visualization, and then in the fields section of the field list, click the Items table name itself.
2. A table is created automatically with the fields Name, Category, Color, and Drawing.

   These are the default fields in this table, as defined in the model on which this report is built.
3. With the table selected, in the fields section of the field list, in the Items table, click the Distributors field.

   It is added to the table.
4. With the table still selected, click the arrow to expand the **Visualizations** gallery,

and then click **Card**.

5. Drag the scroll bar on the right of the card to scroll through the rest of the cards.

   Note the name and image are prominently displayed in each card. This is the result of properties set in the model.

Tiles

To create a tile container

1. With the cards selected, drag the Drawing field from the fields in the card listed in the **Fields** box in the layout section of the field list to the **Tile by** box in the layout section.

   This creates a series of tiles in a container, one tile for each value in the Drawing field. Notice that the card in the tile no longer has a scroll bar. It is filtered to the value in the Drawing field.

   Take a moment to click the different fruits in the tab strip along the top of the tile container. See how the data in the card changes.

2. Drag the sizing handle on the right of the card to make the card taller than it is wide.
3. Hover over the edge of the tile container until the hand appears, and then drag the container to the upper-right corner of the report, next to the title.

   Tip
   You drag a tile container by clicking the edge – not by clicking in the tab strip.

   For tips on selecting, moving, and resizing visualizations, see "Working in the view" in Power View FAQ, Troubleshooting, Tips, and Tricks.

To add to the tile container

1. Select the tile container. Drag the right handle of tile container to make it three times as wide as the card.
2. Click in the tile container, but not the card.
3. In the fields section of the field list, in the Dates table, select the Month Name field.

   This starts a new table in the tile container.

4. In the fields section of the field list, in the Items table, select the Name field.
5. In the fields section of the field list, in the Quantities table, select the Sum of Qty Served field.
6. In the **Visualizations** gallery, click the **Line** chart.

   The line chart has a legend, but you do not need it because the chart contains just one line for the food item on the tile.

7. With the line chart selected, on the **Layout** tab in the **Chart Tools** contextual tab, click **Legend**, and then click **None**.

   Make the chart wider to show the names of all the months.

   You can resize the card and line chart inside the tile container, independent of the container, but you may need to make the container bigger first.

▷To experiment with tiles

1. Click the different foods in the tab strip along the top of the tile container.

   Notice that the card and line chart in the tiles change to show values for the selected food.

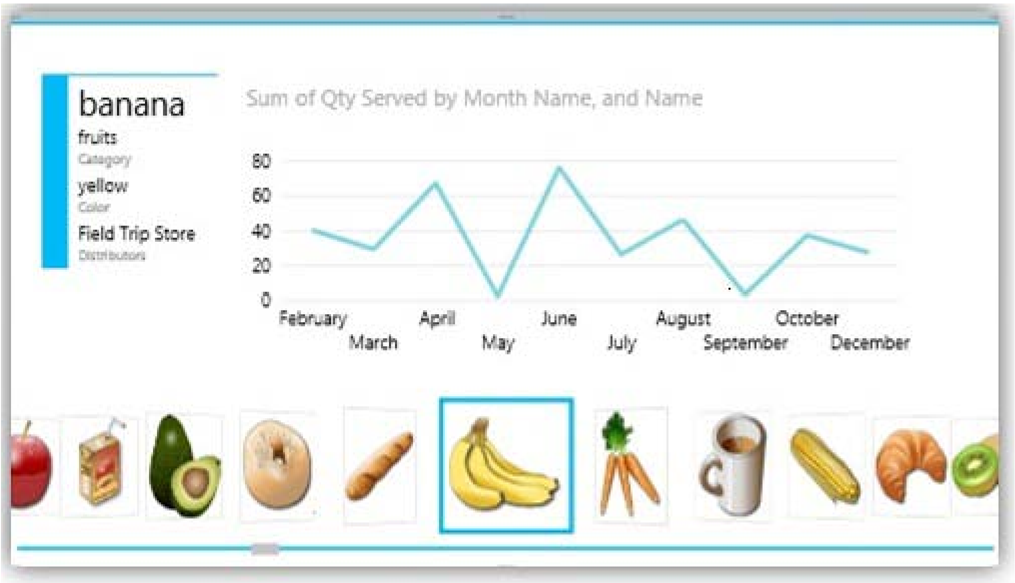
   But also notice that none of the other visualizations in the view change. They are not controlled by the tile container.

2. Click the tile container, and on the **Design** tab in **Tile Visualizations**, click **Cover Flow**.

   You may need to resize the tile container to fit the card and chart.

3. Again, click the food items in the cover flow, or drag the slider, to show values for different food items.

Tile container with cover flow navigation



Compare highlighting and slicer filtering

Now that you have several different visualizations in one view, you can see how filtering and highlighting differ among them.

1. Select a bar in the Category bar chart. Notice that it:
   - Highlights related parts of the other chart.
   - Filters the items in the tile.
   - Has no effect on the slicers.
2. Clear that selection by clicking in the bar chart but not on a bar.
3. Select a category in the Category Drawing slicer. Notice that it:
   - Filters both charts, removing the unrelated bars and columns.
   - Filters the items in the tile.
   - Filters the values in the Drawing slicer.

Add another view

Your report can have more than one view. You can add a duplicate of an existing view or add a new blank view. All the views in a report are based on the same model.

To add a new blank view

- On the **Home** tab, click **New View**.
- Select the title text box and press Delete.

Bubble charts

Scatter charts display two measures and bubble charts display three.

To create a bubble chart

1. Click the new blank view to start a new visualization.
2. In the fields section of the field list, in the Items table, select the Category field.
3. In the fields section of the field list, in the Quantities table, select the Qty Served YTD (year-to-date) and Sum of Qty Served measures.

   Notice that the numbers in each column are the same, because the quantity consumed year-to-date has no date to reference.
4. Click the **Scatter** chart button.
5. Drag one of the sizing handles to make the scatter chart larger.
6. In the fields section of the field list, in the Quantities table, drag the Sum of Qty Consumed measure to the **Size** box in the **Chart Fields** layout section.

This is a snapshot of the totals. The values on the X and Y axes are the same.

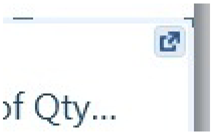
If you add a date component, you can see how the values change over time.

7. In the fields section of the field list, in the Dates table, drag Month Name to the **Play Axis** box in the **Chart Fields** layout section.
8. In the bubble chart, click the play arrow next to Month Name.

   The bubbles move, grow, and shrink as their values change over the months. Notice that the months display as watermarks in the upper-right corner, and the bubbles move to the right as the months pass and the year-to-date quantity increases.

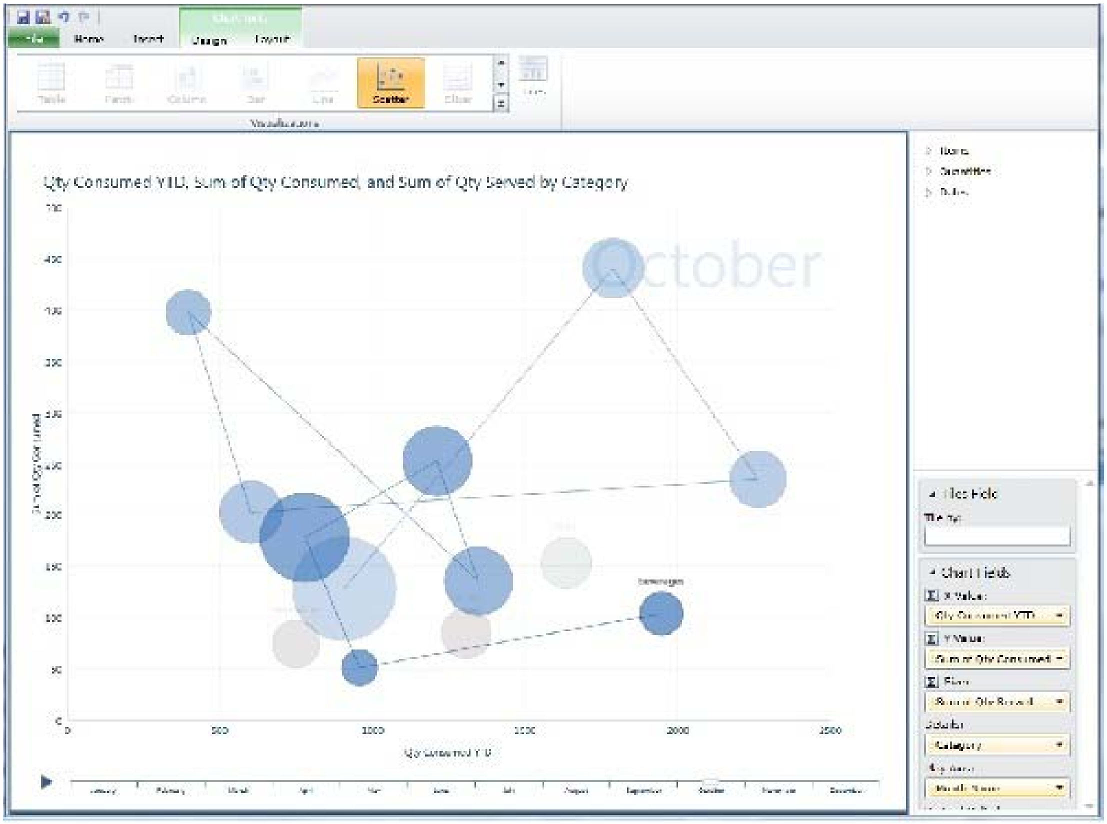
   Note
   If the bubble chart is very small, you may not see the watermark.
9. Click one of the bubbles. This does two things:
   - It draws a line tracing the path of that bubble over time.
   - If there are other visualizations on the view, it highlights the value of the bubble in the other charts on the report and filters the tiles to that value.
10. To see the bubbles even better, you can 'pop out' the chart: hover in the upper-right corner of the chart and click the popout icon.

    Popout button

    Now the chart appears to take up the full view, but only temporarily. The rest of the report is still the same underneath, and when you click the popout icon again, the chart returns to its spot in the report.
11. With one bubble selected and the chart popped out, click the play arrow again.
12. Watch as the bubble traces its path.

    Bubble chart with play axis and data labels, tracing the path of beverages



13. To return the bubble chart to its original size, click the popout icon again.
14. Drag the bubble chart to the upper-left corner of the view.

Add category group colors to a scatter or bubble chart

Sometimes grouping the bubbles in a scatter or bubble chart by category makes the information clearer.

To add category group colors to a bubble chart

1. Drag the Name field from the fields section of the field list to the Details box in the layout section of the field list, replacing the Category field there.

   Now the bubble chart has one bubble for every item. It makes a busy chart.

2. Drag the Category field from the fields section of the field list to the Color box in the layout section of the field list.

   Now the bubbles are colored depending on the category to which they belong.

3. Click a category name in the legend.

   Notice that it highlights the items in that category in the bubble chart.

Small multiples

With tiles and slicers, you can see a series of charts filtered to a specific value. In this tutorial, the tiles filter the data to show the value for one of the foods. The slicers filter the report for one or more categories of food.

With small multiples, you can see data in a series of small charts side by side.

To create small multiples

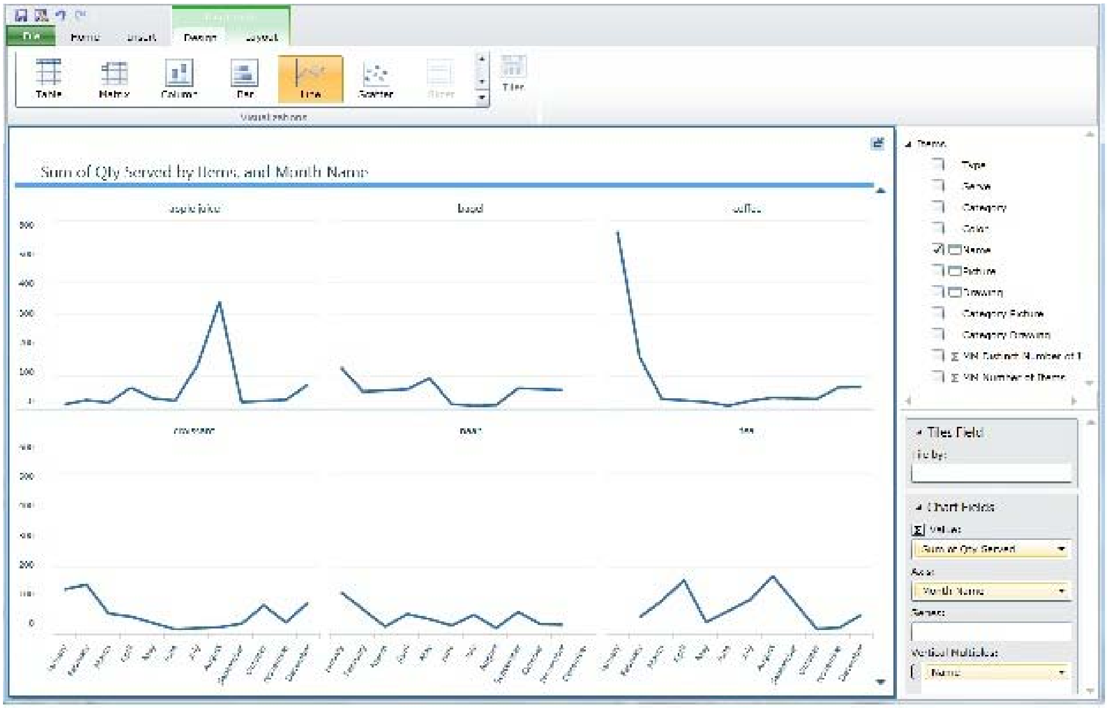
1. Click the blank part of the view next to the bubble chart to start a new visualization.
2. In the fields section of the field list, in the Dates table, click the Month Name field.
3. In the fields section of the field list, in the Quantities table, click the Sum of Qty Served field.
4. In the **Visualizations** gallery, click the **Line** chart.

   This line chart shows quantities for all foods by month.
5. Drag the left and lower sizing handles to make the chart bigger.
6. In the fields section of the field list, in the Items table, click the Name field and drag it to the **Vertical Multiples** box in the layout section of the field list.

   By default, it creates a visualization that displays nine multiples at a time. They are a little crowded.
7. With the small multiples chart selected, on the **Layout** tab in the **Chart Tools** contextual tab, click **Grid**.
8. Drag to select three multiples across and two multiples tall.

   Now you see six multiples at a time, so you can see the individual multiples better.
9. To see them even better, pop out the chart: hover in the upper right corner of the chart and click the popout icon.

   Small multiples popped out to fill the view



Values: Sum of Qty Served

Axis: Month Name

Vertical Multiples: Name

The small multiples now fill the entire design space. Again, this has no effect on the other visualizations in the view.

10. To return it to its original size, click the popout icon again.
11. Drag the small multiple container to the lower-right corner of the view.

View and visualization filters

Like slicers, view filters filter the whole view, but not other views in a report. One advantage of view filters is that they don’t take up any space on the view. Another is that advanced filters provide much more flexibility: For example, you can search for parts of values, rather than just a whole value.

Visualization filters are similar to view filters but they filter individual visualizations on a view, rather than the whole view.

▶ To create a view filter

1. On the **Home** tab, click **Filters Area.**
2. In the fields section of the field list, in the Dates table, click Date and drag it to the filters area.

   In **Basic Filter Mode**, you can select individual dates.
3. Click the **Advanced Filter Mode** button to the right of Date.
4. Click the arrow next to **is on or after**, and then click **is on or before**.

5. Click the calendar and click August 31, 2011.
6. Click **apply filter**.

   This removes dates after August 2011.

▶To create a visualization filter

1. Hover over the vertical multiples and then click the filter icon in the upper-right corner of the visualization.

   Notice that the fields in the visualization are already in the **chart** filters area.
2. In the filters area, click Sum of Qty Served.
3. Drag the right end of the slider so the highest value is 400.

   Notice that the Values axis for the multiples has now changed. The highest number is now 400.

   In the bubble chart, the Y value, also Qty Served, is unchanged – the maximum is still 600.

View your report

You can preview your report, and you can view it in presentation mode. In both modes, the report is still interactive. When you click bars or columns in the charts, it filters and highlights the rest of the report. However, the field list and ribbons are absent, so you cannot add fields and create visualizations, or move or resize existing visualizations. Also, your interactions will not be saved with the report.

▶To preview your report

1. On the **Home** tab, click **Reading Mode**.

   Notice that the filters area is still visible.
2. Click the icons in the lower-left corner or the numbers in the lower-right to move from one view to the other.
3. Try some of the interactions you've tried earlier in the tutorial:
   - Click bars, columns, or bubbles in the charts.
   - Click different foods in the tile container.
   - Click an item in a slicer.
4. From here, you can return to design mode or go to full-screen mode with the buttons in the upper-left corner. Click **Edit Report**.

▶To present your report

1. On the **Home** tab, click **Full Screen**.

   Notice that the filters area is still visible.

2. Try some of the interactions you tried earlier in reading mode.
3. Click ESC to leave full-screen mode.

Save your report

▶ To save your report

1. On the **File** menu, click **Save** or **Save As**.

   The **Save As** dialog box contains the location of the model on which your report is built. You can change that location, saving the report to a subfolder, for example, but you must save the report to the same SharePoint server as the model.
2. Give the report a unique name.

   Tip
   We recommend a name that identifies the model on which the report is built.
3. Keep the **Include an image of each view in its current state as a preview for other users (in PowerPivot Gallery and other applications)** check box checked. For more information about these preview images, see Choose Whether to Save View Preview Images.

Export your report to PowerPoint

You can export your report to PowerPoint. It has the same interactivity that it has in reading and full-screen modes.

▶ To export to PowerPoint

1. On the **File** menu, click **Export to PowerPoint**.
2. Browse to where you would like to save the new PowerPoint file. You can save it anywhere, as long as it can connect to the report stored on the SharePoint server.
3. Open the PowerPoint presentation.

   Each Power View view has been added to a separate slide. In PowerPoint design mode, each is static representation of the view.

   Note
   If you didn't choose to save a preview image when you saved the report, then you don't see a static representation of the view in PowerPoint design mode.
4. Make any changes you want in the PowerPoint presentation.

   Note

You can resize Power View views in PowerPoint, but you can't edit them.

5. In PowerPoint, click **Slide Show**.
6. In the lower-right corner of the slide, click **click to interact**.

   Now you can interact with the view in the same way that you can in reading and full-screen modes.

For more information, see Creating, Saving, Exporting, and Printing Power View Reports.

Next Steps

Now that you've completed this tutorial, you are ready to try using your own data in Power View. To learn more about building data models for use with Power View, try Tutorial: Optimize a Sample PowerPivot Model for Power View Reports.

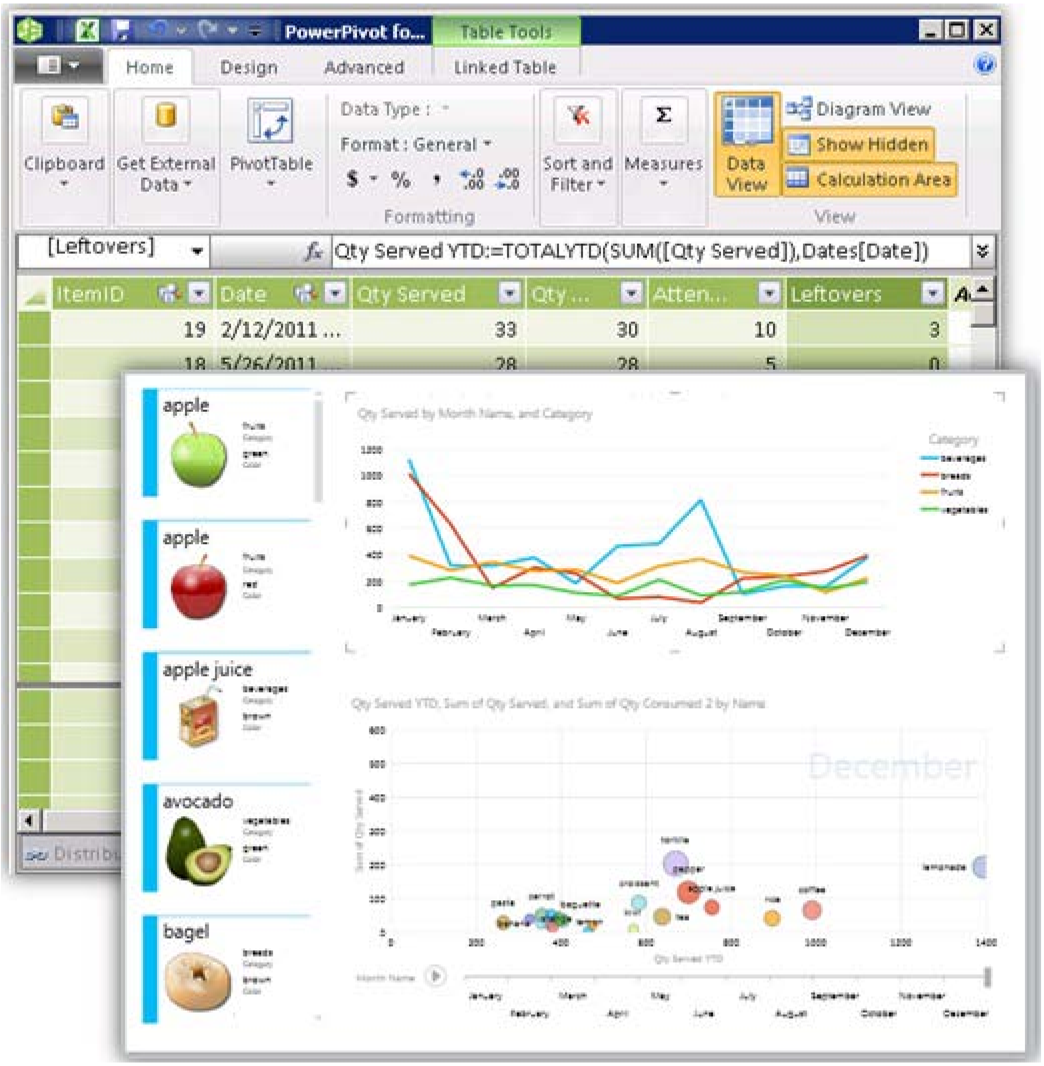
See Also

Project Crescent (SSRS)

Tutorial: Optimize a Sample PowerPivot Model for Power View Reports

In this tutorial, you modify an Excel workbook in PowerPivot for Excel to make a small but functional model that you can use as a basis for sample Power View reports. Everything you do in this tutorial in PowerPivot, you can also do in tabular models that you create in SQL Server Data Tools (SSDT). In fact, you can do much more in SQL Server Data Tools (SSDT). For more information, see Tabular Modeling (SSAS).

The PowerPivot file and Power View report based on it



Qty Served YTD:=TOTALYTD(SUM([Qty Served]),Dates[Date])

ItemID	Date	Qty Served	Qty ...	Atten...	Leftovers
19	2/12/2011 ...	33	30	10	3
18	5/26/2011 ...	28	28	5	0

Prerequisites

To do this tutorial:

- Install PowerPivot for Excel.
- Have a computer that meets the System requirements for Project Crescent.
- Download and install Power View samples, including the image files, from the Download Center. For more information, see Power View and PowerPivot HelloWorldPicnic Samples for SQL Server 2012 and Images for Power View and PowerPivot HelloWorldPicnic Samples in SQL Server 2012 Samples .

Note

HelloWorld_Picnic is a tiny sample data model that demonstrates how to set data model properties to best present that data in Power View reports. In the scenario, the data is used to plan and track food and drink for picnics.

In This Article

Before you start the tutorial

Before you can start the tutorial:

- Save the HelloWorldPicnicModelPowerPivotTutorialRTM.xlsx file to a SharePoint document library or PowerPivot Gallery.
  - If you save it in a SharePoint document library, you create a shared data source that points to the HelloWorldPicnicModelPowerPivotTutorialRTM.xlsx file that you saved in the SharePoint document library.
  - If you save it to a PowerPivot Gallery, it doesn't need the shared data source. You can open Power View directly from the gallery.
- Save the images to the associated SharePoint document library. See more information about the Images for Power View and PowerPivot HelloWorldPicnic Samples

To create a shared data source (RSDS) file to point to the PowerPivot (XLSX) file

1. In Shared Documents, create a HelloWorldPicnicSQL2012 folder, and save the HelloWorldPicnicModelPowerPivotTutorialRTM.xlsx file there.
2. In the HelloWorldPicnicSQL2012 folder, click the **Documents** tab, click the arrow

next to **New Document**, and then click **Report Data Source**.

If there is no **Report Data Source** option on the **New Document** tab, then someone with adequate permissions on the SharePoint site needs to add that content type to the site. For more information, see Add Report Server Content Types to a Library (Reporting Services in SharePoint Integrated Mode).

3. In the **Name** box, type **HelloWorldPicnicRSDS**.

   Note the RSDS file extension.

4. In the **Data Source Type** box, click **Microsoft BI Semantic Model for Power View**.
5. The **Connection string** for an XLSX file is the full URL to the file, including the file name. For example:

   http://<myserver>/Shared%20Documents/HelloWorldPicnicSQL2012/HelloWorldPicnicModelPowerPivotTutorialRTM.xlsx

6. For **Credentials**, click **Windows authentication (integrated) or SharePoint user**.
7. Click **Test Connection**.

   You see a message that the connection test was successful.

8. Click **OK**.

To save the images

1. Create an Images folder in the HelloWorldPicnicSQL2012 folder. For example:

   http://<myserver>/Shared Documents/HelloWorldPicnicSQL2012/Images

2. In the Images folder, save the images that you downloaded from the Download Center. For more information, see the Prerequisites section at the beginning of this tutorial.

To start PowerPivot from the PowerPivot Gallery

1. From the PowerPivot Gallery, click the image of the file HelloWorldPicnicModelPowerPivotTutorialRTM.xlsx.

   The workbook opens in SharePoint.

2. Click **Open in Excel**.
3. Click the PowerPivot tab, and then click PowerPivot Window.

To start PowerPivot from Shared Documents

1. From Shared Documents, click the dropdown arrow next to HelloWorldPicnicModelPowerPivotTutorialRTM.xlsx, and click **Edit in Microsoft Excel**.
2. Click the PowerPivot tab, and then click PowerPivot Window.

In Excel and PowerPivot

Create linked tables in PowerPivot

▶

1. In the Excel window, select all the columns and rows in the **Items** worksheet.
2. On the **Home** tab, click **Format as Table**, and then pick a format for the table. You can make the style anything you like.
3. On the **PowerPivot** tab, click **Create Linked Table**.

   Make sure it contains the correct columns and rows and that **My table has headers** is checked, and then click **OK**.

   The table opens on a new tab in PowerPivot.
4. Right-click the tab and rename it **Items**.
5. Repeat this three times to create linked tables in PowerPivot for Distributors, Quantities, and Dates.
6. Click **Save**, and you now have a model—not much of a model, but a model nonetheless.

   Note
   You're back in the Excel window after you save.

In Power View

Start Power View

▶**From Shared Documents**

- In the Shared Documents folder, click the arrow next to HelloWorldPicnicRSDS.rsds, the shared data source you created earlier in this tutorial, and then click **Create Power View Report**.

▶**From PowerPivot**

- In the PowerPivot Gallery, click the **Create Power View Report** icon in the upper-right corner of the HelloWorldPicnicModelPowerPivotTutorialRTM.xlsx band.

Start a report

Power View opens to a blank view. The field list on the right contains the tables you created in PowerPivot: Dates, Distributors, Quantities, and Items.

1. Expand the Items table.

   There are the fields from the table in PowerPivot.

2. Click the Items table name itself.

   Nothing happens. We'll come back to this later.

3. Click the Name field.

   You now have a table with one column, the Name column.

   Note, though, that now all the other tables are grayed out, unavailable. Why?

In PowerPivot: Create relationships

The other tables are grayed out because there are no relationships between the tables in the model.

To create relationships in PowerPivot

1. In the PowerPivot Window on the **Home** tab, click **Diagram View**.

   You see the tables in your model, with no relationships between them.

2. Drag the tables so you can see them all in the window. Place the Quantities table in the center, as it is central to the model.

3. Drag from the ItemID field in the Items table to the ItemID field in the Quantities table.

   PowerPivot creates a one-to-many relationship between the two tables. It doesn't matter which way you drag. The one and many sides of the relationship are the same. For more information about one-to-many relationships, see Guide to table relationships (from Microsoft Access Help).

4. Drag these fields to create other relationships between:

   - The Date field in the Dates and Quantities tables.
   - The DistribID field in the Items and Distributors tables.

5. Save the Excel (XLSX) file, thus saving your model.

In Power View

When you make changes to the model, you need to refresh the report. There are two Refresh buttons to choose from:

- The **Refresh** button in the Power View Quick Access toolbar refreshes the data, if the data in the model has changed.
- The **Refresh** button in Internet Explorer refreshes the model, but deletes any changes you’ve made since you last saved.

▶

1. Click the Power View **Refresh** button.

   Note that nothing changes.
2. Click the Internet Explorer **Refresh** button.
3. In this case we haven’t made any changes worth saving, so click **Leave this page**.

   The field list is still there but the view is blank.
4. Expand the Items table, and drag the Name field to the view.

   Note that the other tables are not grayed out. Note, too, that there is only one “apple” value.
5. Expand the Quantities table and drag the Qty Served field to the Name table.

   Now there are many “apple” values. The Qty Served values are not being aggregated.
6. Try scrolling down the list.

   Note that the scroll bar gets smaller as Power View retrieves more values. Power View retrieves values only as needed, to speed performance.
7. In the layout (lower) section of the field list, click the dropdown arrow next to the Qty Served field and then click **Sum**.

In PowerPivot: Set reporting properties

Time to set reporting properties to make reports based on this model work better.

▶

1. In the PowerPivot Window on the **File** menu, click **Switch to Advanced mode**.
2. Go to the Items table.
3. On the **Advanced** tab, click **Table Behavior**.
4. For **Row Identifier**, click **ItemID**.

   Note that now the rest of the items in the dialog box are available. Until you have selected a row identifier, they are all grayed out.
5. For **Keep Unique Rows**, click Name.

   Thus, even if there are two identical values in this column in two different rows, Power View doesn't aggregate them if they have different ItemID values. For example, “apple”: There are two different “apple” values in the Name field. Power

View no longer aggregates them, as it did earlier.

6. For **Default Label**, click Name.
7. Click **OK**.
8. Click **Default Field Set**.
9. Add Name, Category, and Color, and then click **OK**.

   These fields are the defaults that are added to a view in Power View if you click the table name rather than an individual field.
10. Save the Excel (XLSX) file to save your model.

In Power View: Create a table and cards with default values

1. Click the Internet Explorer **Refresh** button.

   Because you haven't saved the report yet, you lose the few changes you have made.
2. Click **Leave this page**.
3. Expand the Items table.

   Note the icon next to the ItemID and Name fields indicating that they are unique fields and the default label for this table.
4. Drag the Name field to the view.

   Note that there are now two apple values.
5. Click the **Undo** button.
6. Click the Items table itself.

   This creates a table with the fields you identified as the default fields for this table: Name, Category, and Color.
7. With this table selected, click the dropdown arrow in the Visualizations Gallery and click Card.

   Note that the values from the Name field are displayed prominently, because Name is the default label field for the Items table.
8. Click someplace on the blank view so the cards are not selected.
9. In the fields (upper) section of the field list, expand the Distributors table.

   Note that it contains only the Distributor and DistID fields.
10. Save the Power View report someplace on the SharePoint site.

In PowerPivot: Consolidate tables and add images

We can consolidate the single field from the Distributors table into the Items table to clean up the model.

To create a calculated column

1. In the PowerPivot window in the Items table, right-click the **Add Column** column and click **Insert Column**.
2. Select the new column, right-click, click **Rename**, and rename it **Distributors**.
3. In the Excel formula box, type

   ```
   =RELATED(Distributors[Distributor Name])
   ```

   Note that the formula autocompletes as you type. You can use autocompletion to make sure you have the right syntax.

   The Distributor name is now in the Items table, so report creators no longer need the Distributors table or the DistID field in the Items table.
4. Right-click the DistID column, and then click **Hide from Client Tools**.

   This hides the column in Power View, but you can still see it in PowerPivot.
5. Right-click the Distributors table tab and click **Hide from Customer Tools**.

Add images to the model

Before you can add images to the model, download them from the Microsoft Download Center and save them to the SharePoint site. For more information, see the Before you start section.

1. Add a calculated column to the Items table and call it Image Name.
2. Paste in this formula:

   ```
   =IFERROR(IF(FIND("apple",[Name])>0,
   SWITCH(Left([Color],1),"b","applejuice","g",[Name]&
   "g","r",[Name],[Name]),[Name]),[Name])
   ```

   This column makes the image names correspond to the item names in the Name field.
3. Add another calculated column and name it Photo. Paste in this formula, and replace <servername> with your server name:

   ```
   ="http://<servername>/Shared
   Documents/HelloWorldPicnicSQL2012/Images/rs_crescent_picnic_i
   mage_" & [Image Name] & ".png"
   ```

   Each item has an associated photo. This formula concatenates the name in the

Image Name column with the URL where the images are located, plus the .png file extension.

4. Add another calculated column and name it Drawing. Paste in this formula, and replace <servername> with your server name:

   ```
   ="http://<servername>/Shared
   Documents/HelloWorldPicnicSQL2012/Images/rs_crescent_picnic_d
   rawing_" & [Image Name] & ".png"
   ```

   Each item has an associated drawing, too.

5. Add another calculated column and name it Category Drawing. Paste in this formula, and replace <servername> with your server name:

   ```
   ="http://<servername>/Shared
   Documents/HelloWorldPicnicSQL2012/Images/rs_crescent_picnic_d
   rawing_" & [Category] & ".png"
   ```

   Each category has an associated drawing.

6. Add one more calculated column and name it Category Photo. Paste in this formula, and replace <servername> with your server name:

   ```
   ="http://<servername>/Shared
   Documents/HelloWorldPicnicSQL2012/Images/rs_crescent_picnic_i
   mage_" & [Category] & ".png"
   ```

   Each category has an associated photo, too.

7. Save the Excel (XLSX) file to save your model.

   Note that you are back in the Excel window, and the calculated columns don’t appear. They are only in PowerPivot.

Let’s see how it looks in Power View.

In Power View: Try to add images to a table

▶

1. Click the Internet Explorer **Refresh** button.

   Because you have saved the report, you don’t lose the changes you have made.

   Note that there are only three tables now—the Distributors table is gone.

2. Expand the Items table.

   Note the new fields, including the Distributor field.

3. Drag the Photo column to the table on the view.

   Uh oh! That’s no photo. That’s just the URL. Back to the model.

In PowerPivot: Identify image URLs

1. Click the Items table, then the **Advanced** tab.
2. Check the **Image URL** box for all four URL fields.
3. Click **Table Behavior**.
4. Set the **Default Image** to **Drawing**, and then click **OK**.
5. Click **Default Field Set**, and add **Drawing** to the **Default fields** set.
6. Click **OK**.
7. Save the Excel (XLSX) file to save your model.

In Power View: Add images, dates, and numbers to a table

To add images

1. Click the Internet Explorer **Refresh** button.

   You lose the most recent changes you made.
2. Add the Drawing field to the table.

   There are the images!

Now let's look at dates and numbers.

To add dates and numbers

1. Click the blank space in the view to start a new visualization.
2. Expand the Dates table and drag the Month Name field to the view.
3. From the Items table, add the Category field to the same table.

   Note that none of the charts in the Visualizations Gallery are available. A chart needs at least one aggregated value.
4. In the Quantities table, click the dropdown arrow next to the Qty Served field, and click **Add to Table as Sum**.

   Now that the table has a field with a sum, the charts are available.

   The Qty Served field contains whole numbers. Power View does not aggregate them by default because whole numbers might be in a field that shouldn't be aggregated, such as a key field. You can change that default behavior in the model.
5. Click **Line Chart**.

You should see:

- Qty Served in Values.
- Month Name in Axis.
- Category in Series.

If not, drag the fields to the right boxes.

But something is wrong: The months start with April, August, and December, and end with September. They're alphabetical, not chronological.

Back to the model.

In PowerPivot: Default aggregates, dates, tooltips, and calculated measures and columns

Calculated columns calculate a result and store it for every row in a table. Calculated measures are calculated on the fly, depending on the context—where they are being calculated in a PivotTable or in a Power View report. For more about calculated columns and measures, see Build Formulas for Calculations.

To set month order

1. In the PowerPivot window, go to the Dates table.
2. On the **Design** tab, click **Mark as Date Table**.
3. Select the Date field and click **OK**.

   Note
   The values in the Date field must be unique.

4. Select the Month Name column, and on the **Home** tab, click **Sort by Column** and then **Sort Month Name** by the Month Number column.
5. Select the Month Number column, right-click, and click **Hide from Client Tools**.
6. Repeat steps 4 and 5 for the Day Name and Day Number fields.

To set default aggregates

1. In the PowerPivot window, go to the Quantities table.
2. Select the Qty Served, Qty Consumed, and Attendees columns.
3. On the **Advanced** tab, click **Summarize by**, then click **Sum**.

To set a description

- Right-click the Qty Served column, click **Description**, and type:

  How much we served.

  This text will appear as a tooltip over the Qty Served field in the fields (upper) section of the field list. You can add a tooltip to every table and field.

To create a calculated column

1. In the Quantities table, insert a column and rename it **Leftovers**. In the Excel formula box, type:

   ```
   =[Qty Served]-[Qty Consumed]
   ```

   PowerPivot autocompletes the column names as you type.
2. Select the Leftovers column and on the **Advanced** tab, click **Summarize by**, then click **Sum**.

To create a calculated measure

1. Still in the Quantities table, click in any cell in the lower part of the table, below the dividing line.

   This area is called the measure grid. You can hide or show it with a button on the **Home** tab. You can put measures in any cell in any column.
2. Paste the quantity served year-to-date formula in the Excel formula box:

   ```
   Qty Served YTD:=TOTALYTD(SUM([Qty Served]),Dates[Date])
   ```
3. Save the Excel (XLSX) file to save your model.

In Power View: Create a bubble chart

1. Click the Internet Explorer **Refresh** button.

   What has changed?

   - The line chart refreshes to put the month names in order.
   - The Quantities table in the field list has new numeric fields such as Qty Served, with little calculator symbol next to them. The Qty Served field now appears two times—one is aggregated as a measure, and the other isn't aggregated. You could hide the unaggregated column in the model.
   - Also in the Quantities table, the new Qty Served YTD measure has the little calculator symbol next to it.
2. In the fields section of the field list, hover over the Qty Served field.

   The tooltip appears.
3. Create a table with the Name, Qty Served, Qty Served YTD, and Leftovers fields.
4. In the **Visualizations** gallery, click **Scatter** chart.

   Make sure the Qty Served YTD field is in the **X Value** box.

5. Drag the Month Name field from the fields section of the field list to the **Play Axis** box in the layout (lower) section of the field list.
6. Click the play arrow.

   Note that the bubbles gradually move from left to right, as the Qty Served YTD values increase.

In Excel: Validate the model

You can also validate that your model has published correctly by creating and testing a slicer on your data in Excel. You save your XLSX file to the SharePoint site, which you have already done. You create a PivotTable and add slicers to it in Excel. In the XLSX file on the SharePoint site, you click a slicer, and it sends a query that tests the data connection. If the slicer is successful, that proves that the embedded PowerPivot model has been successfully deployed and that the credentials are configured correctly.

▶

1. In the PowerPivot window, on the **PowerPivot** tab, click **PivotChart**, then **New Worksheet**.

   Note that the new worksheet displays the PowerPivot Field List on the right.
2. From the Dates table, drag Month Name to **Slicers Horizontal**.
3. From the Items table, drag Color to **Slicers Vertical**, and Type to **Slicers Horizontal**.
4. From these tables:
   a. Items table: drag Serve to **Legend Fields**.
   b. Items table: drag Category to **Axis Fields**.
   c. Quantities table: drag Leftovers to **Values** (note that it is automatically aggregated as a Sum).
5. Right-click the chart and click **Change Chart Type**.
6. Click **Stacked Column.**
7. Click the different values in the slicers to filter the chart.

The slicers work, so the PowerPivot model embedded in the Excel workbook is successfully deployed and the credentials are configured correctly.

Summary

You’ve seen how you started with raw data, and then:

- Created relationships.
- Set reporting properties, identifying:
  - Unique values.
  - Default labels.

    - Default images.
    - Default fields in a table.
- Added calculated columns and a measure.
- Consolidated tables.
- Set number format.
- Bound date name fields to number fields for order.
- Brought in images.

And you've seen how those changes changed the way the data appeared in the report.

Next Steps

Now that you've completed this tutorial, you can use the model you have created to try Tutorial: Create a Sample Report in Power View in Power View Samples for SQL Server 2012 and SQL Server 2012 Samples Readme.

See Also

Project Crescent (SSRS)