

Authoring Reports Using Microsoft SQL Server 2008 Reporting Services Report Builder 2.0

Quickstart guide

Published: February 2009

***Abstract***

This document provides step-by-step instructions for building several types of reports using SQL Server 2008 Reporting Services applications, and sample data. The document focuses on the new SQL Server 2008 Report Builder 2.0 application.

Contents

[Introduction 1](#_Toc221699831)

[Before You Begin 1](#_Toc221699832)

[Lab Scenario 2](#_Toc221699833)

[Exercise 1: Getting Started and Getting Familiar with Report Builder 3](#_Toc221699834)

[Exercise 2: Pre-Requisite Exercise—Creating a Shared Data Source 6](#_Toc221699835)

[Exercise 3: Table and Matrix Wizard—Data Sources, Data Sets, and Building a Matrix 7](#_Toc221699836)

[Exercise 4: Enhancing Your Report with Richly-Formatted Text 19](#_Toc221699837)

[Exercise 5: View Your Report from the Server 22](#_Toc221699838)

[Exercise 6: Enrich Your Report Using Charts 24](#_Toc221699839)

[Chart Wizard and Styles 30](#_Toc221699840)

[Exercise 7: Enrich Your Data Visualization Using a Gauge 32](#_Toc221699841)

[Exercise 8: Modify the Tablix to Show Two Parallel Dynamic Groups 42](#_Toc221699842)

[Exercise 9: Manual Creation of a Tablix to Match Exercise 3 (Wizard Tablix) 47](#_Toc221699843)

[Exercise 10: Use Query, Parameter, and Filter to Limit the Tablix Data 52](#_Toc221699844)

[References and Where to Get Help 59](#_Toc221699845)

Information in this document, including URL and other Internet Web site references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place, or event is intended or should be inferred. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

The names of manufacturers, products, or URLs are provided for informational purposes only and Microsoft makes no representations and warranties, either expressed, implied, or statutory, regarding these manufacturers or the use of the products with any Microsoft technologies. The inclusion of a manufacturer or product does not imply endorsement of Microsoft of the manufacturer or product. Links are provided to third party sites. Such sites are not under the control of Microsoft and Microsoft is not responsible for the contents of any linked site or any link contained in a linked site, or any changes or updates to such sites. Microsoft is not responsible for webcasting or any other form of transmission received from any linked site. Microsoft is providing these links to you only as a convenience, and the inclusion of any link does not imply endorsement of Microsoft of the site or the products contained therein.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Copyright © 2007 Microsoft Corporation. All rights reserved.

Microsoft registered trademarks or trademark(s) of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

V1

Version 1.2

# Introduction

This document is a step-by-step instruction manual for building several reports using SQL Server 2008 Reporting Services applications and sample data. The focus is on using the new SQL Server 2008 Report Builder 2.0 application. See the **Required Environment** section for the software required to complete these exercises.

## Before You Begin

#### Estimated time to complete this lab

* Basic Exercises 1 – 5: 30 Minutes
* Advanced Exercises 6 – 9: 45 minutes

#### Objectives

After completing this lab, you will be able to:

* Launch the new report designer application **Report Builder 2.0**.
* Create a data source and a data set.
* Design a new report using Report Builder 2.0.
* Preview your report from the application and from Report Manager.
* Enrich your report using the richly format-able text box.
* Enrich your report using a chart.
* Enrich your report using gauge.
* Enhance your Tablix and create two dynamic column groups.
* Add end-user power by adding dynamic filtering.

#### Required Environment

This lab requires the following:

* Microsoft SQL Server Reporting Services Report Server, for publishing reports.
* Microsoft SQL Server 2008 Reporting Services Report Builder 2.0:

<http://www.microsoft.com/downloads/details.aspx?familyid=9F783224-9871-4EEA-B1D5-F3140A253DB6&displaylang=en>

* Microsoft SQL Server 2008 Adventureworks Sample Database:

<http://www.codeplex.com/MSFTDBProdSamples/Release/ProjectReleases.aspx?Released=18407>

# Lab Scenario

You are a report author for AdventureWorks—a sporting goods retailer. You have a requirement to develop a new report that displays product sales information. The report is to be designed to show aggregated sales data using two-level dynamic groupings on both the row and column axis. You also need to create an advanced report that leverages some of the new design capabilities introduced with Reporting Services 2008.

A Reporting Services 2005 table enables quick and easy groupings of basic data. A Reporting Services 2005 matrix enables complex groupings of data, including nested groups and inner static members within dynamic columns. In Reporting Services 2008, a new data region called **Tablix**, combines the table and matrix features, combining the best of both. Therefore, the new Tablix functionality allows you to build reports that include the following:

* Multiple parallel row/column members at each level
* A mix of dynamic or static members on both rows and columns
* Optional omission of member headers
* Arbitrary nesting on each axis

# Exercise 1: Getting Started and Getting Familiar with Report Builder

In this exercise, you will:

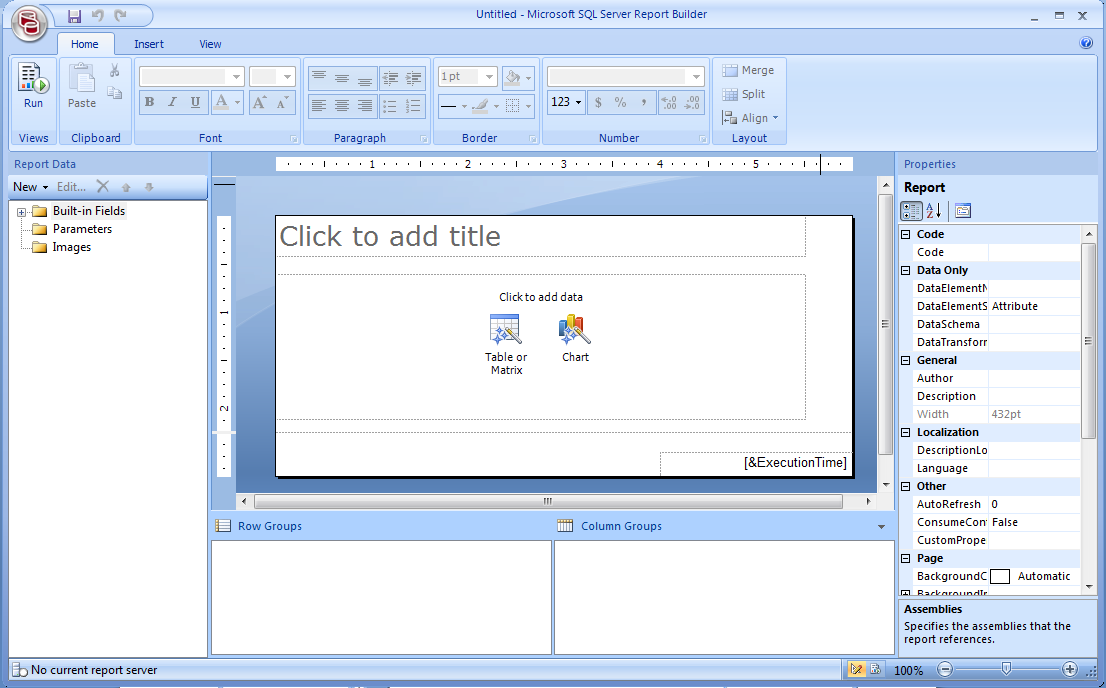
* Launch Report Builder 2.0 (RB2).
* Get started with a blank design surface for creating a new report.
* Review the various key areas of the RB2 user interface (UI).

Launching Report Builder 2.0

To launch the Report Builder, perform the following steps:

1. On the Start menu, click **Programs, then click Microsoft SQL Server 2008 Report Builder 2.0.**
2. **Click Report Builder 2.0.**

Report Builder 2.0 launches with a blank report, as shown in Figure 1.



**Figure 1: Creating a New Blank Report Using Report Builder 2.0.**

|  |
| --- |
| **Note** |
| If your screen does not have all the same windows open in RB2, in the Menu bar, click the **View** tab and select all four items. |

Exploring Key Areas of RB2.0

The following table describes several key areas of the Report Builder 2.0 User Interface.

| **Area** | **Image** | **Notes** |
| --- | --- | --- |
| **Report Data** |  | This panel provides access to create and use data sources, data sets, parameters, and images—all the key pieces of information you would want to use in your report creation. |
| **Properties** |  | This panel lists and edits the detailed properties of the current report item (textbox, line, and so on) or report section (header, footer, and body).  The most common properties are also exposed in property dialogs when you right-click an object. |
| **Grouping** |  | This section enables you to manage the properties of all of your groups for Tablix report items (table, matrix, list). You can easily create groups using the window, or dragging fields to the grouping window. |
| **Server status in status bar** |  | This area reports the current report server/connection status. The server can be changed from the Report Builder icon menu and choosing **Options**. |
| **Zoom control in status bar** |  | Report Builder 2.0 supports zooming in design and run mode. |
| **Context menus** |  | RB2 consists of a rich set of context menus, for accessing almost every feature, including the virtual space around the report. |
| **Snap Lines** |  | Design Surface Snap lines help align objects. |

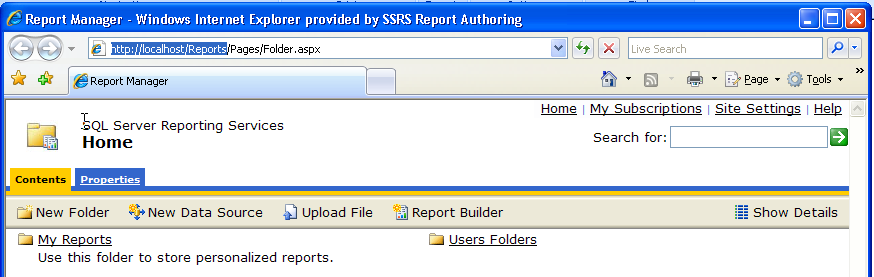
# Exercise 2: Pre-Requisite Exercise—Creating a Shared Data Source

In this exercise you will create a shared data source on the report server, which can be used for multiple reports. Utilizing shared data sources on the server allow centralized management of connectivity to data sources.

Creating a Shared Data Source

To complete this exercise, perform the following steps:

1. Open Internet Explorer and type in the URL of the report manager of [**http://localhost/Reports**](http://localhost/Reports) (or your server URL) and press **ENTER**.
2. Click **New Data Source** in the Contents page of the Report Manager, as shown in the **Figure 2**.



**Figure 2: Report Manager**

1. On the **New Data Source** page, perform the steps:
   * In the **Name** field**,** enter **LocalAdventureWorks.**
   * For **Data Source Type,** select **Microsoft SQL Server.**
   * Enter the following in the **Connection String.**

Data Source=localhost;Initial Catalog=AdventureWorks2008

* + Near the bottom, click to select **Windows Integrated Security.**

1. Click **Apply.**
2. Minimize Internet Explorer to return to it later to preview reports from the server.

# Exercise 3: Table and Matrix Wizard—Data Sources, Data Sets, and Building a Matrix

In this exercise, you will:

* Design a new report using the **Table or Matrix** wizard.
* Specify the report data source. The data source to be used is the **AdventureWorks** sample database that you can download from Codeplex. (See references section at the end of the document).
* Specify the Report Data Set. The data set is generated from a custom query that returns sales data, including **ProductCategory**, **SubCategory**, **Product**, and **Year**.
* Specify fields and groupings.

This exercise demonstrates how to create a report showing **Product Category** and **Sub Category** for the row axis, and **Region** and **Order Year** for the column axis.

* Select report layout of style and colors.

This exercise assumes that you have completed Exercise 1.

Designing a New Report Using the Table or Matrix Method

To design a new report, follow these steps:

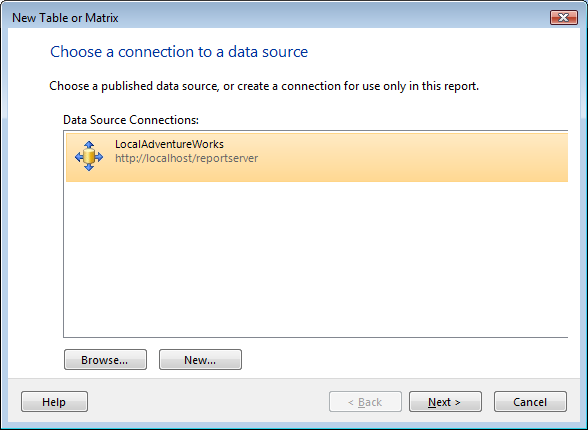
1. Open the Report Builder 2.0.
2. In the new blank report, click the **Table or Matrix** wizard link in the center of the screen.

The wizard walks you through creating a Table or Matrix. The process includes the creation of the required data source and data set.



|  |
| --- |
| **Note** |
| You can also access the wizard from the toolbar by selecting **Matrix Wizard** from the **Matrix** Menu as well as the **Table Wizard** from the **Table menu,** both in the **Insert** tab. The icon in your new report disappears after you run the wizard the first time. However, you can re-run the wizard at any time from the Insert ribbon. |

1. At this point, you may briefly see the server dialog informing you the application is connecting to the server. The new Table or Matrix Wizard dialog appears. Select the existing data source connection **LocalAdventureWorks**, and click **Next**.



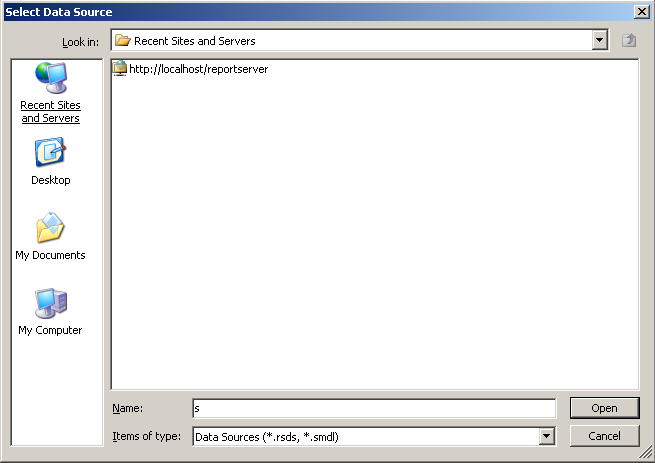
**Figure 3: New Table or Matrix Wizard**

If you cannot see the shared data source you previously created, click **Browse** to find the data source on the server.

Specifying the Data Source

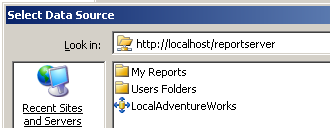
To browse to your specific data source, perform the following steps:

1. Double-click the server name: [**http://localhost/reportserver**](http://localhost/reportserver)



**Figure 4: Browse Server to Data Source**

1. Double-click the **LocalAdventureWorks** shared data source you previously created.

****

**Figure 5: Select Your Data Source**

1. Click **Next** to move to the Query Design step of the wizard.

Specifying the Data Set

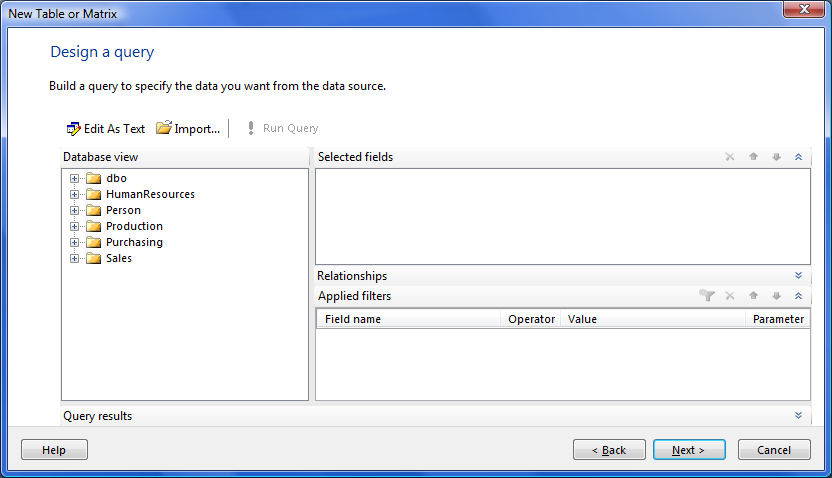
Here, the data set is generated from a custom query that returns sales data, including **ProductCategory**, **SubCategory**, **Product**, and **Year**.

It follows that the next step in the wizard is to create a query (or a data set) for the matrix.

To streamline the process, perform the following procedure:

1. Copy and paste the query instead of building a query manually.

|  |
| --- |
| **Note** |
| With the new Query Designer in Report Builder 2.0, you can explore the schema of your data source to make it easier to build queries. You can also import the query syntax from a file using the **Import** functionality. |

****

**Figure 6: Query Designer**

1. Click**Edit as Text** to switch the query designer to text mode.
2. Select all the query text below, copy it, and paste it into the query editor.

SELECT

PC.Name AS ProductCategory,

PS.Name AS SubCategory,

DATEPART(yy, SH.OrderDate) AS OrderYear,

'Q' + DATENAME(qq, SH.OrderDate) AS OrderQtr,

SUM(SD.UnitPrice \* SD.OrderQty) AS SalesAmount,

Sales.SalesTerritory.[Name] AS Territory,

Sales.SalesTerritory.[Group] AS Region

FROM

Production.ProductSubcategory AS PS JOIN

Sales.SalesOrderHeader AS SH JOIN

Sales.SalesOrderDetail AS SD ON SH.SalesOrderID = SD.SalesOrderID JOIN

Production.Product AS P ON SD.ProductID = P.ProductID ON PS.ProductSubcategoryID =

P.ProductSubcategoryID JOIN

Production.ProductCategory AS PC ON PS.ProductCategoryID = PC.ProductCategoryID JOIN

Sales.SalesTerritory ON SH.TerritoryID = Sales.SalesTerritory.TerritoryID

WHERE

(SH.OrderDate BETWEEN '1/1/2003' AND '12/31/2004')

GROUP BY

DATEPART(yy, SH.OrderDate),

PC.Name,

PS.Name,

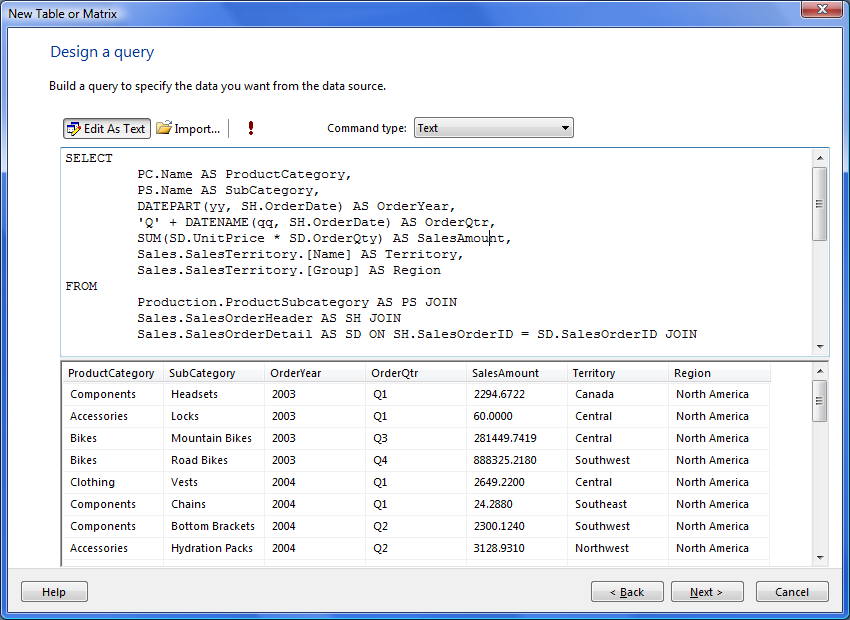
'Q' + DATENAME(qq, SH.OrderDate),

PS.ProductSubcategoryID,

Sales.SalesTerritory.Name,

Sales.SalesTerritory.[Group]

At this point, we suggest you execute the query to ensure everything is working—you can verify this by seeing rows populated in the results pane at the bottom. To execute the query, press .



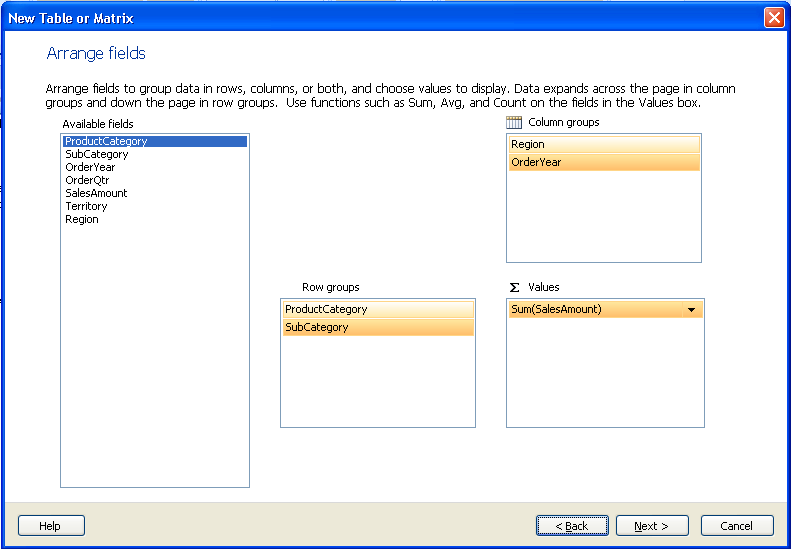
**Figure 7: Text-based Query Design in the Wizard**

1. Click **Next** to continue to arrange the fields.

Specify the Fields and Groupings

In continuation with the Report Builder wizard, specify the fields and groupings in the following steps:

1. To arrange the fields(**Column Groups**, **Row Groups**, and **Values**), do the following:
   1. Drag **SalesAmount** to the **Values** definition box.
   2. Drag **ProductCategory** and **SubCategory** to the **Row groups** definition box.
   3. Drag **Region** and then **OrderYear** to the **Column groups** definition box.



**Figure 8: Arrange fields in the Wizard to Specify Groups and Values**

|  |
| --- |
| **Note** |
| If you drag a field to the wrong location, you can fix this by dragging it from one definition box to another, or to the white/blank space in between them to remove it. |

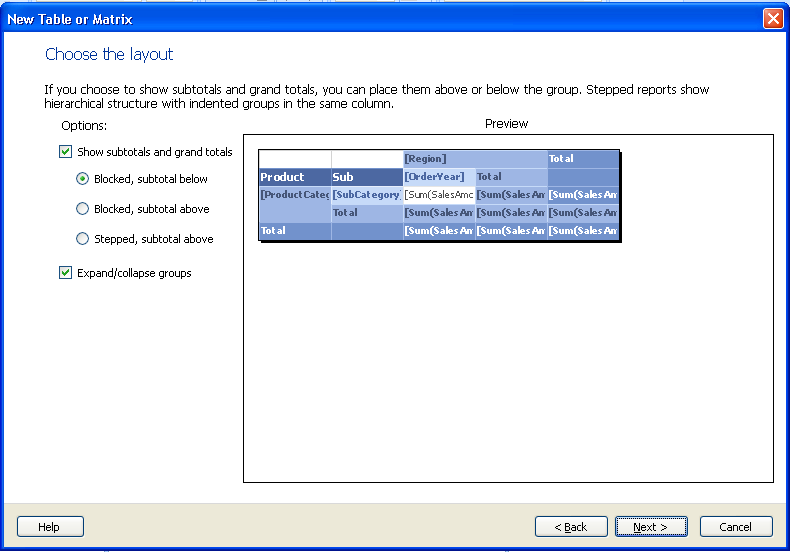
1. Click **Next** to continue.

Selecting the Layout and Colors

1. Select the visual layout.

For this exercise, use the default **Blocked, subtotal below** layout. We recommend clicking the other options to understand the options available with the wizard, based on the preview.

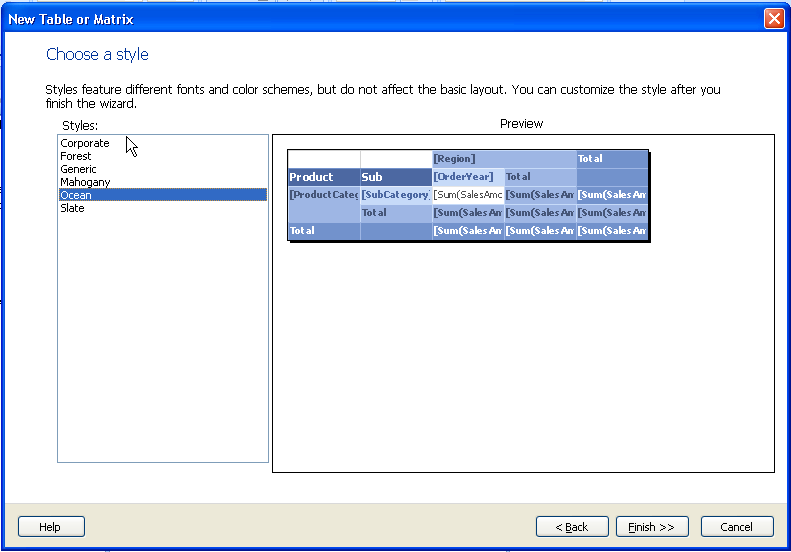
|  |
| --- |
| **Note** |
| The **Expand/Collapse groups** checkbox builds the report with group headers and totals shown, but the detail rows will be hidden by default. However, the report will show **+/-** symbols on the groups to allow you to expand and see the detailed rows while running or previewing the report. |



**Figure 9: Select the Visual Layout Type.**

1. Click **Next** to proceed to the Style selection.
2. Select the style.

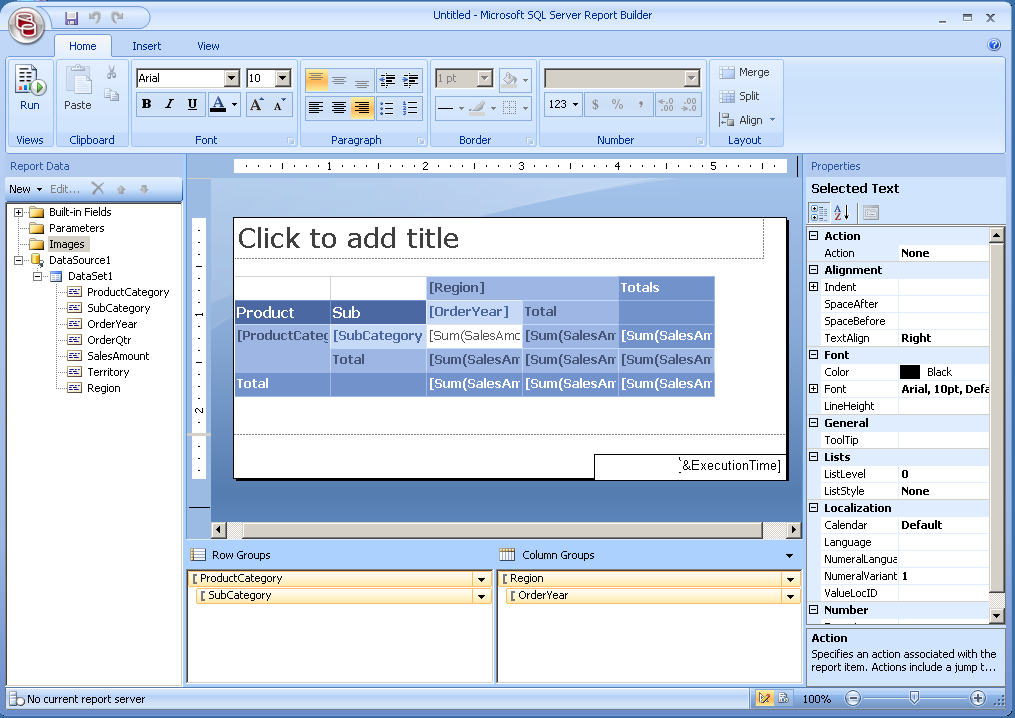
For the purpose of the exercise, we use the default **Ocean** style. From the wizard screen, it is possible to modify the colors and the fonts even after wizard is complete.



**Figure 10: Select a Color Style.**

1. Click **Finish** to complete the wizard and view your new report.

Your report will look like the following figure:

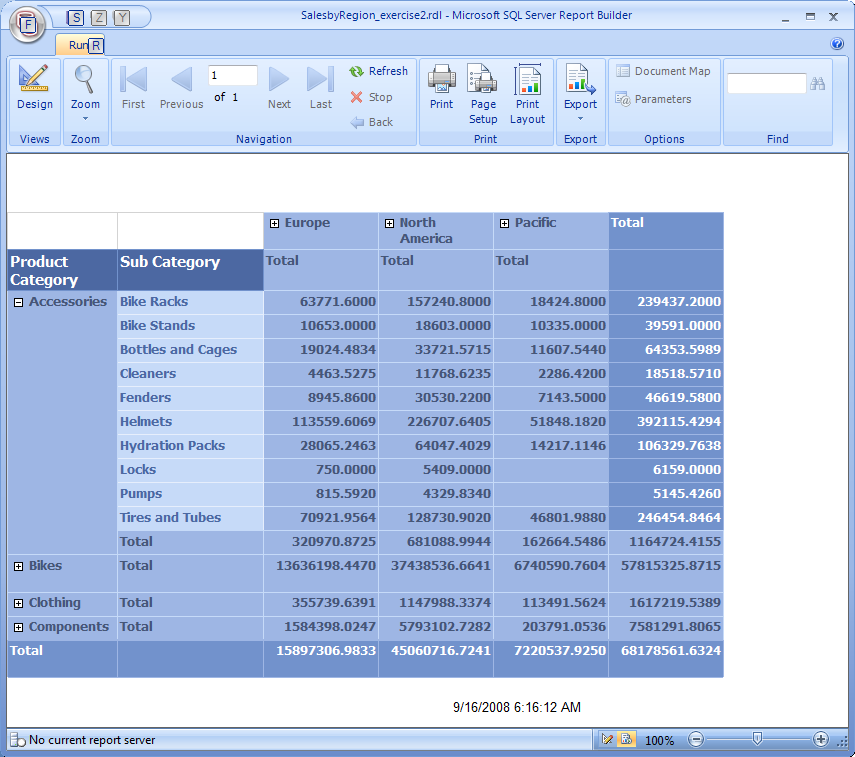


**Figure 11: Completed Wizard**

|  |
| --- |
| **Note** |
| * Notice your data source and data set in the Report Data window on the left. * As you click on the each field in the grouping window, it is selected in the **Tablix**. The column and row selectors provide an additional visual cue of how elements are grouped. For example, if you click on **[ProductCategory]** in the **Row Groups** area, you can see the grouping indicators such as below:     **Grouping Indicators**   * Try the Zoom slider on the status bar at the lower right corner of the screen. |

To run the report, perform the following procedure:

1. Click the **Home** tab, and click **Run** button or press **F5.**

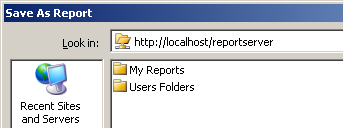
****Your report will look similar to the following.

**Figure 12: Completed Wizard with Expanded** Accessories **Product Group**

1. Use the **Expand and Collapse** **(+/-)** functionality in the **Accessories** group to explore the report.

|  |
| --- |
| **Note** |
| * The **Export** button on the Run ribbon includes various options for export formats.     Export Menu   * Try the run time Zoom functionality on both the Ribbon and the lower right status bar. |

1. To switch back to design mode, Click **Design** or press **F8**.
2. To save your report to the server, do the following:
   1. Click the Report Builder icon  **, then click Save As**.
   2. Double-click **My Reports** in the folder list.
   3. Enter the file name **SalesbyRegion\_ex3.rdl**.
   4. Click **Save**.

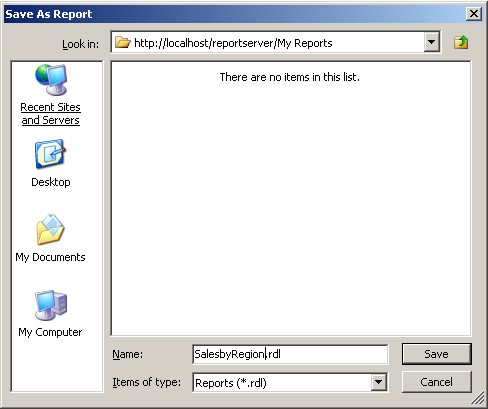


If you are unable to see **My Reports**, perform the following steps to save the report:

1. Click **Recent Sites and Servers** in the left panel.
2. Double click <http://localhost/reportserver>.
3. Double-click **My Reports**.

If **My Reports** was not enabled on the server, you can save to the root of your report server.

1. Enter the file name **SalesbyRegion.rdl**.
2. Click **Save**.



**Figure 13: Save As Dialog**

# Exercise 4: Enhancing Your Report with Richly-Formatted Text

In this exercise, you will:

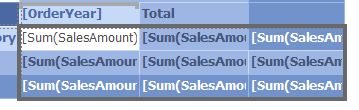
* Enhance the look of your report by using rich formatting on a report title.
* Use place holders for dynamic text.

This exercise assumes that you have completed Exercise 3

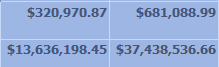
Enhancing the Look of the Report

To enhance the look of the report, follow these steps:

1. Open the **SalesbyRegion\_ex3.rdl** report created in the previous exercises, if it is not already open.
2. To set the sum fields to show currency format, do the following:
3. Click the upper-left sum field, the **Sum(SalesAmount)** under **[OrderYear]**,
4. Hold down the **SHIFT** key, and click **Sum(SalesAmount)** in the lower-right corner to select all of the sum fields.



1. Click the currency format button on the ribbon.

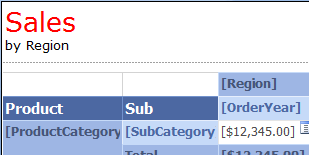


You can also click and format each field individually. When you run the report, you will see the currency format. When in design mode, you can resize your columns wider so numbers do not wrap.

|  |
| --- |
| **Note** |
| The ribbon allows you to toggle the design time view of placeholders from sample values to actual placeholder text. |

1. Click in the textbox at the top of the report showing **Click to Add Title**, and enter a title **Sales by region**.
2. Select the word **Sales** you just typed in, and use the Ribbon to format the font color to red.
3. Select the words **by region** and use the Ribbon to change the font size to 10.
4. Click between the words **Sales** and **by** and press **ENTER**.

This creates a line break with the small text forced to a second line. You can resize the title textbox and move the **Tablix** down. The design view of the report looks like the following figure.



**Figure 14: Formatted Report Title**

Using Placeholders for Dynamic Text

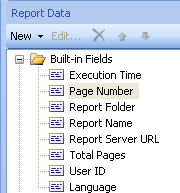
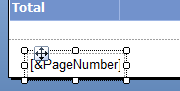
To use placeholders, perform the following steps:

1. Click in the textbox in the lower-right corner of the page footer showing **[&ExecutionTime]**.

This text box contains a **Place Holder** specifically for the built-in field **ExecutionTime**, which prints the time at which the report was executed on the report.

1. Type **Executed:** before the **[&ExecutionTime]** so the whole string is **Executed: [&ExecutionTime]**.
2. Expand the **Built-in fields** list in the Report Data window, and drag **Page Number** to the lower left corner of the report footer.

A new text box is created with the **[&PageNumber]** place holder.

**** 

1. Click the right edge of the new textbox and resize the box to the right to make the textbox wider. You can also use **SHIFT+ Right Arrow** to resize the box.



1. In the text box, type **of** after the **[&PageNumber]** place holder, and from the Report data window, drag **Total Pages** to the end.
2. Select the text and make it bold. You can make the selected text made bold by pressing **CTRL**+**B**,similar to Microsoft Office Applications.



1. To run the report, click the **Home** tab, and click **Run** or press **F5.**

The placeholder now changes to actual values. For example, here, the **[&ExecutionTime]** placeholder is now a date and time.

By default, the report layout appears in a small/collapsed size. If you expand several product categories (**+**), and switch to the **layout** mode, the page total reflects the increased number of pages. To change the layout, click the **Print Layout** button in the ribbon.



Click **Design** or press **F8** to return to Design view.

|  |
| --- |
| **Note** |
| This exercise only gives you an idea of how you can enhance your report with the new text box and the ability to richly format your text. Various combinations of static text (such as **Executed**), placeholders, support for one to many lines, numbered or bulleted lists, paragraph alignment, and the rich font formatting make the new textbox a powerful addition in SQL Server Reporting Services 2008. |

* **Save your report to the server**: To save your report on the server, perform the following steps:



1. Click the Report Builder icon**, then select Save As**.
2. Select the **My Reports** folder, if present, on your local server.
3. Enter the file name ‘**SalesbyRegion\_ex4.rdl**’.

# Exercise 5: View Your Report from the Server

In this exercise, you will:

* View your report from Report Manager using Internet Explorer.

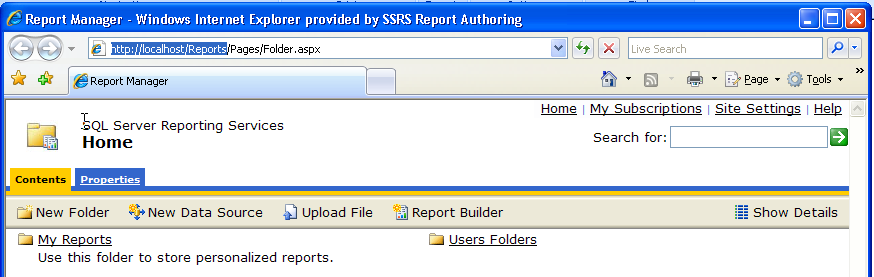
This exercise assumes that you have completed Exercise 4

Viewing the Report from Report Manager

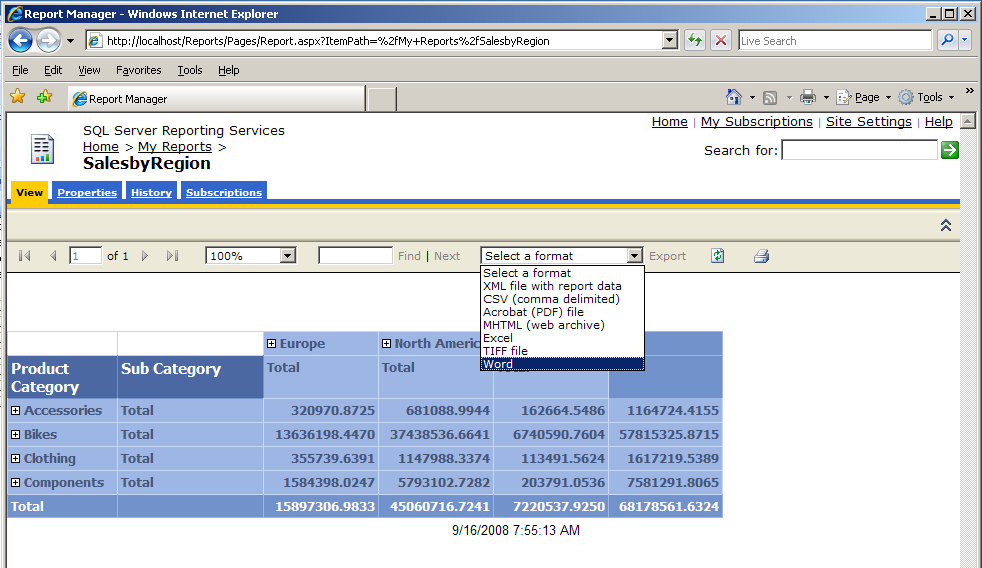
To complete this exercise, follow these steps:

1. Open Internet Explorer and type in the URL to the report manger of [**http://localhost/Reports**](http://localhost/Reports)**,** then press **ENTER**.
2. Click **My Reports** to open the folder.
3. Click the **SalesbyRegion\_ex4** report (or the name you saved it as) .

Your report looks like the way you saw before when you ran it inside Report Builder 2.0.



**Figure 15: Report Manager**



**Figure 16: Viewing the Report in Report Manager**

Note that the same export options in the Report Builder 2.0 application are available in the Report Manager as well as zoom and printing.

The Role Report Manager also enables you to assign permissions and create **Subscriptions** for reports, and can automatically e-mail the report results to required e-mail aliases. This report dispatch can even be scheduled, for example every morning at 8:30.

# Exercise 6: Enrich Your Report Using Charts

In this exercise, you will:

* Add a basic Chart to the Report you previously created.

We will manually build a simple chart rather than use the wizard to get a feel of modifying the report items. SQL Server Reporting Services 2008 introduced new Data Visualization functionality in the form of a richer chart report item, and an all-new gauge report item.

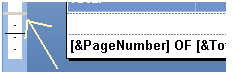
|  |
| --- |
| **Note** |
| The **Data Region** report items such as **Chart**, **Gauge**, and **Tablix** are only allowed in the **Report Body** section, and not in the Header and Footer sections. |

This exercise assumes that you have completed Exercise 4.

Adding a Basic Chart to Previously Created Report

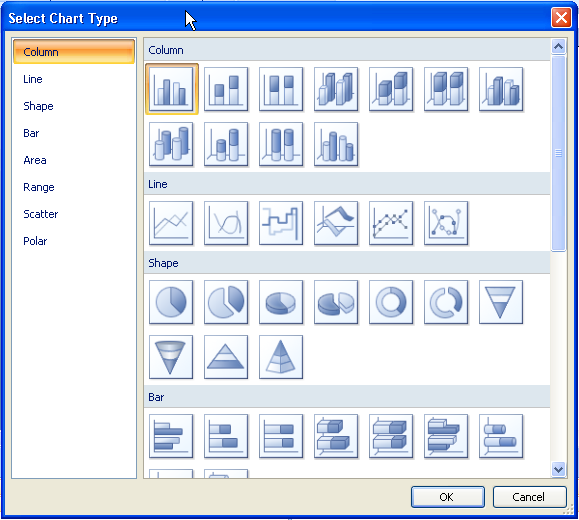
To add a Basic Chart to a previously created Report, follow these steps:

1. Open the previously created report, **SalesbyRegion4\_ex.rdl**.
2. To add a chart report item to the design surface below your existing Tablix, do the following:
3. Create some space at the bottom of your report by dragging the report body sizing handle down in the vertical ruler. This will create a blank space below your Tablix, and will be used to accommodate the chart.



1. In the ribbon, click the **Insert** tab, then select **Insert chart** under **Chart**. The mouse cursor is now in **Insert** mode.
2. Place the pointer to upper-left corner of the new chart, or drag to place and size the new chart.
3. Click immediately beneath the lower-left edge of the Tablix in the Report body.

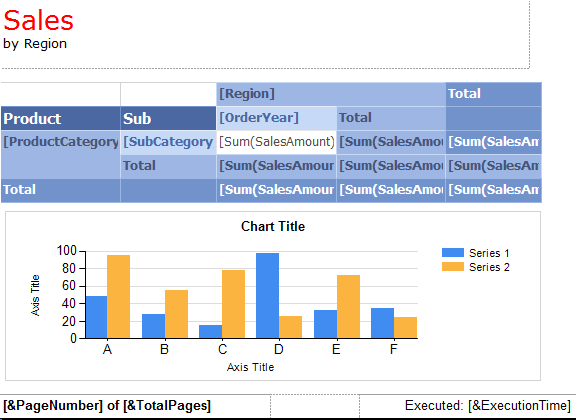
The Select Chart Type dialog box appears.



**Figure 17: Selecting the Chart Type**

1. Select the default column chart in the upper left corner, then click **OK**.

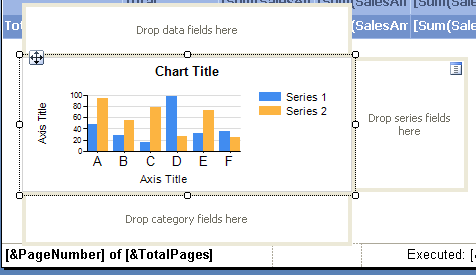
The chart report item is now placed on the design surface. There is no need to select a Data Source and Data Set because the report already has only one, so Report Builder will have set it for you. The Design Time view of your report will look like the following.



**Figure 18: Chart Report Item is Created**

1. To put the chart into edit mode, double-click the chart.

Your Chart report item now looks like the one below, with the field areas visible.



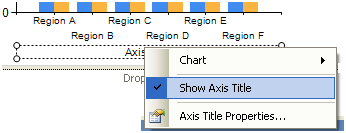
**Figure 19: Chart Showing Field Areas Where You Drag From the Report Data Window**

In the previous Figure, notice that there are three editing areas.

1. Drag and drop fields from the **Report Data** window as follows:

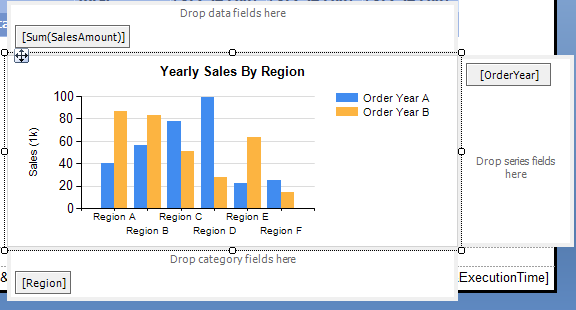
* **Drop data fields here** – Area for fields that represent the numeric **measures** in the chart. In this example, the data field would be the **SalesAmount** field, which represents numeric sales data. Drag this field into the data fields area.
* **Drop series fields here** – Area for individual bars (series) within a category. In this example, if you want to see one bar for each year of sales within a category, select the **OrderYear** series field. Drag this field into the Series area.
* **Drop category fields here** – Area for categories of data. This area is represented by the **A** through **F** labels at the bottom of the chart. In this example, if all series bars should be grouped by **Region**, drag this field into the category area.

1. To change the chart title, double-click the title and enter **Yearly Sales by Region**.
2. To change the vertical axis title, double-click the text box and enter **Sales (1k)**.
3. To remove the horizontal axis title, right-click the axis and clear the **Show Axis Title** check box.



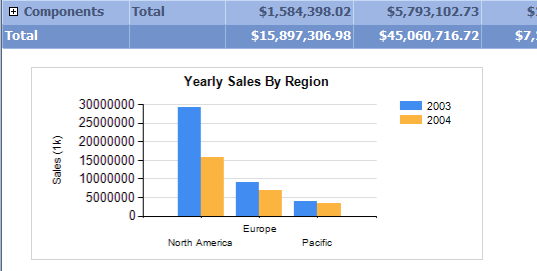
**Figure 20: Turn Off the Horizontal Axis Title**

The chart will look like the following figure:



**Figure 21: Chart Design**

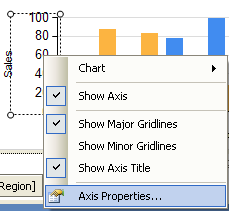
1. To run your report, click the **Home** tab and click **Run** or press **F5.**



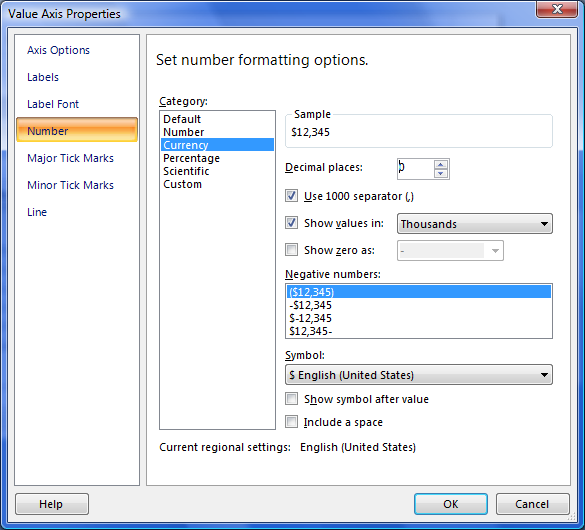
**Figure 22: Previewing the chart**

The result should resemble the chart above. Let's modify this chart further, and format the sales numbers on the left X-Axis for currency.

**To format the Sales numbers in the X Axis for currency, perform the following steps:**

1. Switch back to design mode by clicking **Design** or press **F8.**
2. ****Right-click the numbers of the sales at the X- Axis, then select **Axis Properties…**.

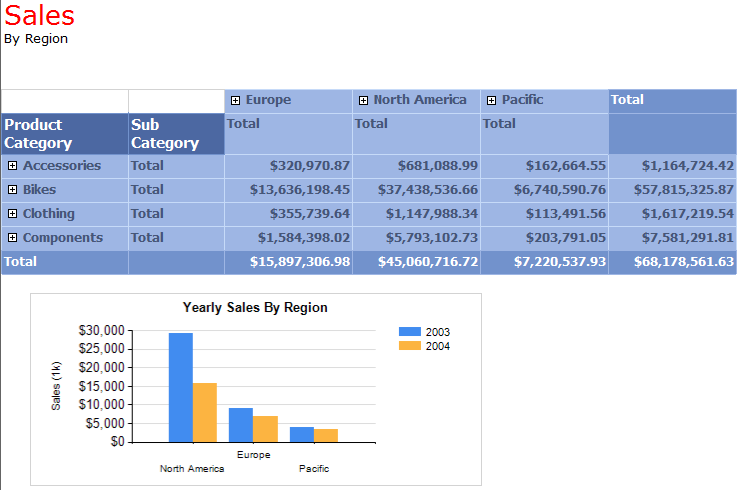
To see the **Axis Properties** option, right-click on the Axis Title instead of the actual Axis Numbers.



**Figure 23: Value Axis properties Dialog**

1. In the Value Axis Properties dialog box, select the **Number** pane and select the **Currency** category.
2. Set **Decimal places** to **0**.
3. Select the **Use 1000 separator** check box.
4. Select the **Show values in** check box, then select **Thousands** from the drop-down list.
5. To run your report, click the **Home** tab, then click **Run** or press **F5.**

Note the X-Axis and the currency format.



**Figure 24: Previewing the Completed Chart**

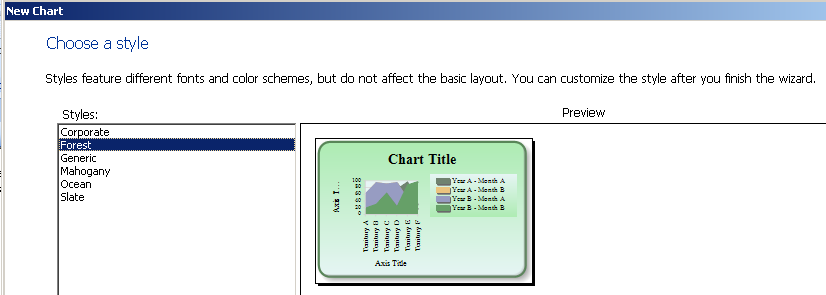
* **Save your report to the server**: To save your report to the server, perform the following steps:



1. Click the Report Builder icon**, then select Save As**.
2. Select the **My Reports** folder, if present, on your local server.
3. Enter the file name ‘**SalesbyRegion\_ex4.rdl**’.

## **Chart Wizard and Styles**

The Chart Wizard enables you to easily create charts. The Chart wizard includes a list of pre-defined chart styles to select from, and it also guides you through creating data sets.



**Figure 25: Sample of the Chart Styles Found in the Chart Wizard**

# Exercise 7: Enrich Your Data Visualization Using a Gauge

In this exercise, you will:

* Add a basic horizontal gauge to the Report you previously created.

The gauge adds a visual queue regarding the sum of sales amount per category, and the way they stack up to a desired goal of 80 million.

|  |
| --- |
| **Note** |
| Remember the design time functionality to zoom in/out, which becomes more useful as your report gets larger and more complex. |

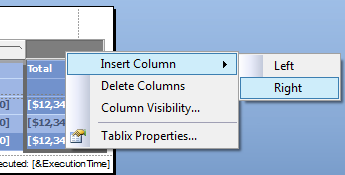
This exercise assumes that you have completed Exercise 6.

Adding a Horizontal Gauge to the Report

To complete this exercise, follow these steps:

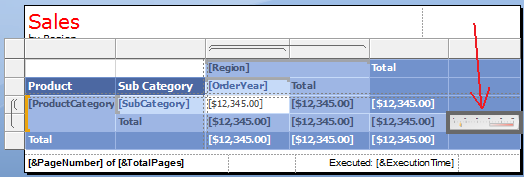
1. Open the report **SalesbyRegion\_Ex6.rdl** created previously.
2. To add a new column to the right edge of the Tablix, right-click the header and select **Insert Column** and then click **Right**.

|  |
| --- |
| **Note** |
| If you do not see the headers, click anywhere in the Tablix. |

****

**Figure 26: Insert a New Column**

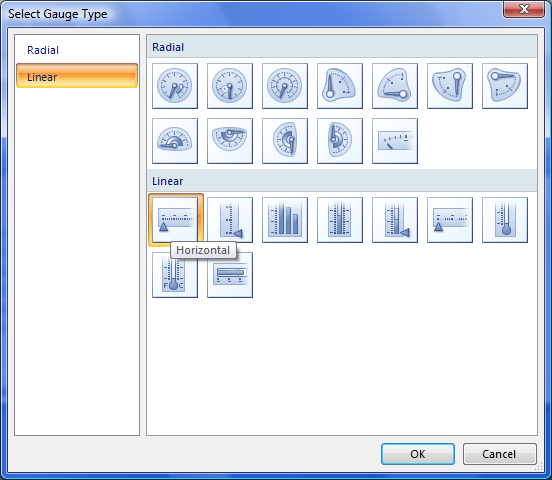
1. In the **Insert** ribbon, click **Gauge** and then click in the last cell of the newly added **Total** row of the **ProductCategory** group.



**Figure 27: Insert a Gauge into the new column on the Total row**

The **Select Gauge Type** dialog appears.

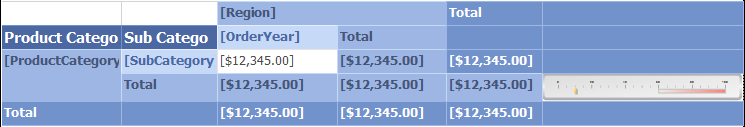
1. Select the first **Linear** type**, Horizontal**, then click **OK.**



**Figure 28: Select Gauge Type**

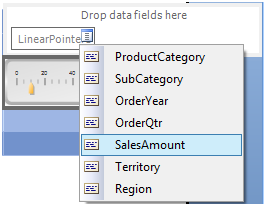
1. To adjust the size of the Tablix and Gauge, drag the row and column wider to view and to click on individual elements within the gauge, such as the rectangular **pointer** and the **scale** of numbers.

The Tablix looks as below.

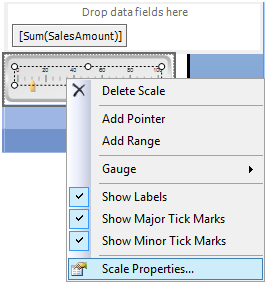


**Figure 29: Resize Gauge**

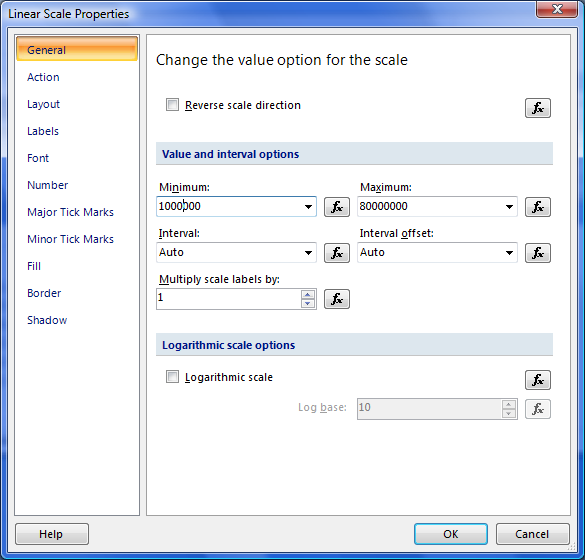
1. To add the Data field in the Guage, click the Field Selector  inside the Gauge and select **SalesAmount**.
2. Click the Gauge so field area is visible, then click the **LinearPointer** area to see the field selector.



1. To set the **Scale Properties**, right-click the scale and select **Scale Properties…**.

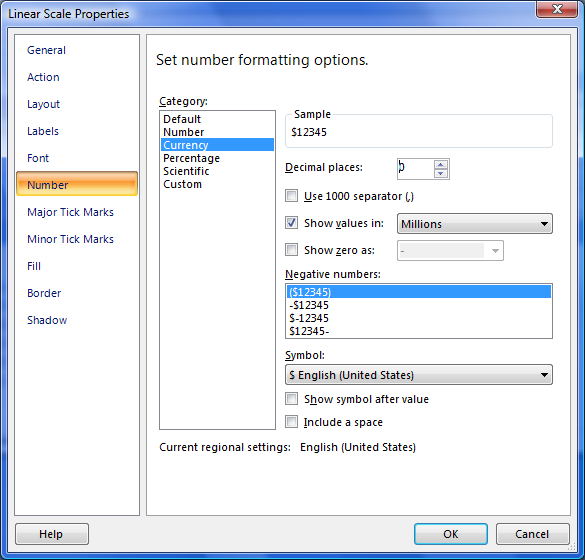


1. In theLinear Scale Properties dialog box, select the **General** category.
   1. Set the **Minimum** scale to **1000000** (one million).
   2. Set the **Maximum** scale t**o 80000000 (**eighty million).



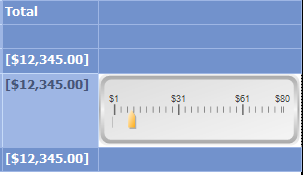
**Figure 30: Linear Scale Properties—Setting the Min/Max**

1. Select the **Number** category, then do the following:
   1. In the **Category** list, select **Currency**.
   2. Set **Decimal places** to **0**.
   3. Select the **Show values in** check box, then select **Millions** from the drop down list.



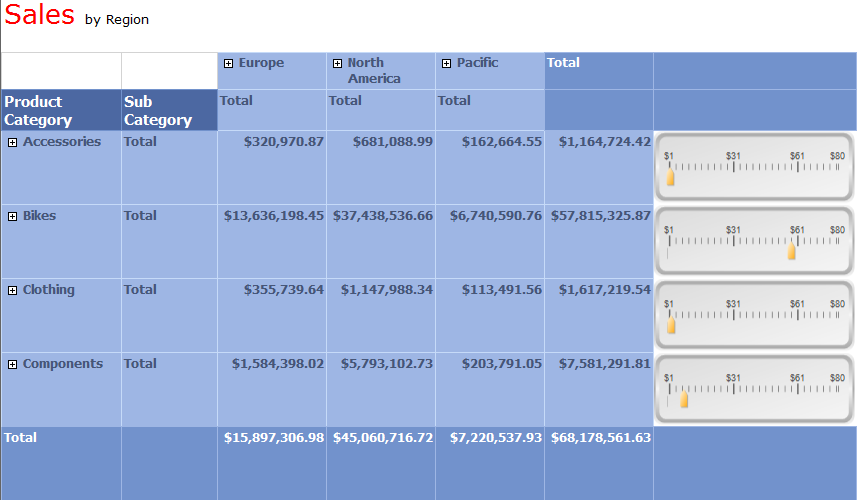
**Figure 31: Currency Settings**

1. Click **OK**.



1. To run your report, click the **Run** tab or press **F5**.

The rendered report will look similar to the second figure.

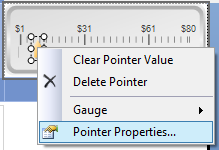


**Figure 32: Previewing with the Gauge**

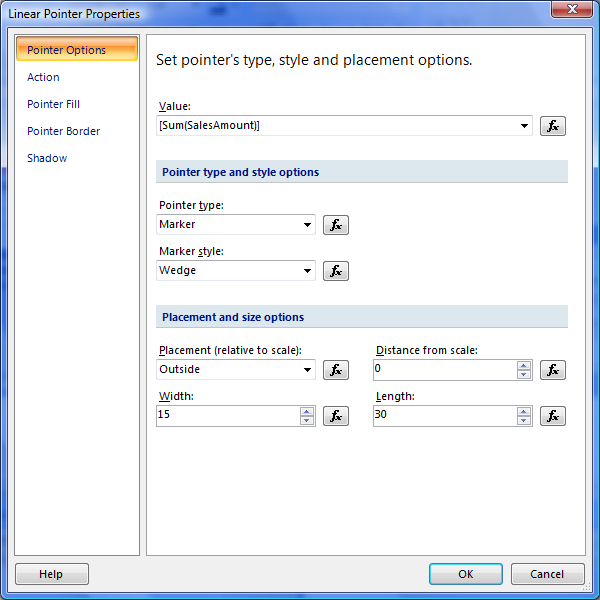
The report now has the visual aid of the gauge to make it easy to understand the relative sales performance of the different categories.

Take it one step further and, using an expression, alter the gauge so the gauge pointer color depends on the sales amount itself. At this point, we should also make the pointers larger for better readability.

1. Right-click the pointer, then select **Pointer Properties.**

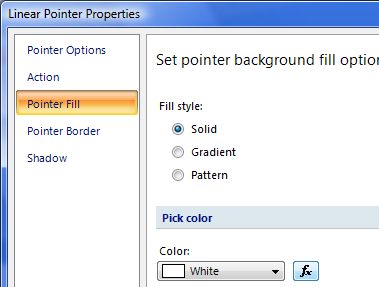


1. In the Linear Pointer Properties dialog box, select the **Pointer Options** category. To enlarge the pointer for easy viewing, do the following:
   1. Set the **Width** of the pointer to **15**.
   2. Set the **Length** of the pointer to **30**.



**Figure 33: Resizing the Pointer**

1. Select the **Pointer Fill** category and do the following:
   1. Select the **Solid** option for **Fill Style**.
   2. To select the color, click the function button  at the **Color** drop-down list.



**Figure 34: Pointer Fill Style and Color**

To relate the color with the position of the guess, perform the following steps:

1. In the Expression dialog, enter a formula that supplies the following logic for pointer color:

Red: SalesAmount < five million.

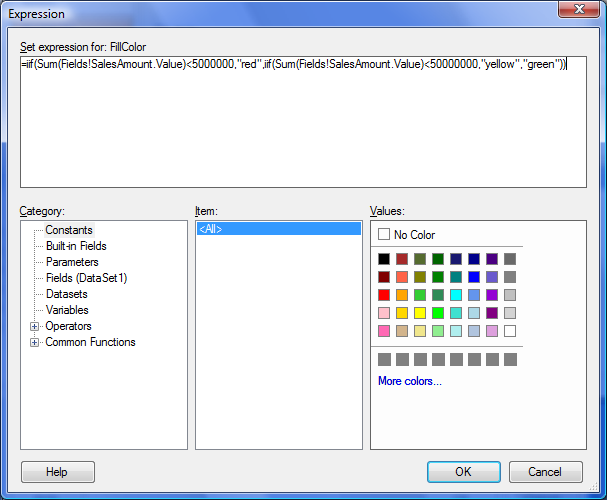
Yellow: SalesAmount > five million but less than fifty million.

Green: SalesAmount > fifty million.

* 1. Delete the default ‘White’ entry.
  2. Copy and paste the following formula into the expression editor.

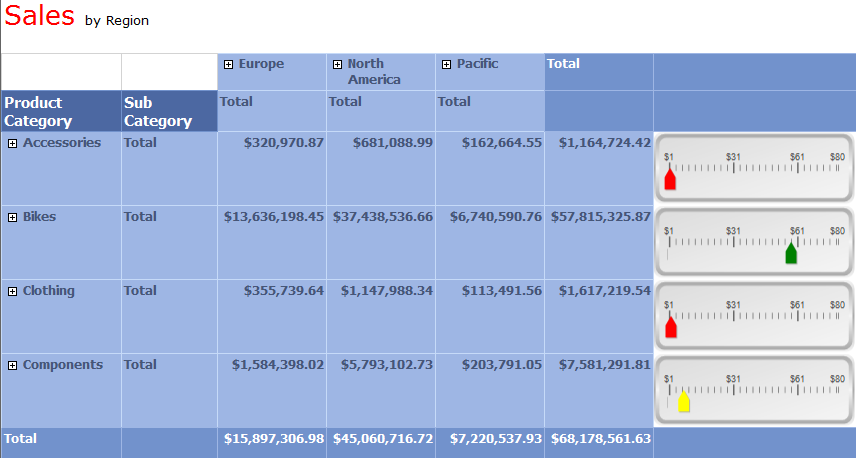
**=iif(Sum(Fields!SalesAmount.Value)<5000000,"red",iif(Sum(Fields!SalesAmount.Value)<50000000,"yellow","green"))**

1. Click **OK**.



**Figure 35: Expression Dialog**

1. Run your report: Go to the ‘Home’ tab, then click the ‘Run’ button or press **F5.**



**Figure 36: Previewing the Completed Gauge with Colored Pointers**

* **Save your report to the server**: To save your report on the server, perform the following steps:



1. Click the Report Builder icon **and then select Save As**.
2. Select the **My Reports** folder, if present, on your local server.
3. Enter the file name ‘**SalesbyRegion\_Ex7.rdl**’.



# Exercise 8: Modify the Tablix to Show Two Parallel Dynamic Groups

In this exercise, you will:

* Modify the Tablix report created in Exercise 3, and enhance its functionality.
* Enhance the functionality by moving the **OrderYear** sub-grouping to be displayed as a new adjacent column, thereby enabling a second independent dynamic group within a single report item—something new for SQL reporting Services 2008.

The group brace symbols are a visual indication of what we will be changing.

|  |  |  |
| --- | --- | --- |
|  | TO |  |

Another subtle hint is the spacing in the Grouping window, with Group1 no longer being indented.

|  |  |  |
| --- | --- | --- |
|  | TO |  |

* Remember you can ‘Undo’ actions using **<CTRL>**+**<Z>** or the Undo button in the Ribbon 

This exercise assumes that you have completed Exercise 7. The images show the Gauge from Exercise 7 but is not required to complete this lesson—just the core Tablix.

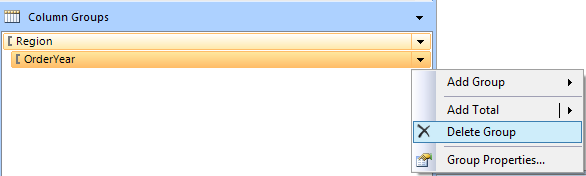
Modifying the Tablix to Enhance Functionality

To modify the Tablix, perform the following steps:

1. Open the report **SalesbyRegion\_Ex7.rdl** you previously created.
2. Delete the **OrderYear** sub group.

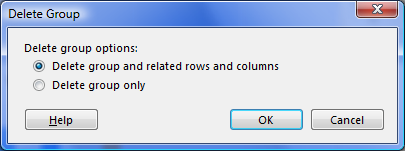
Delete this group to add it as a second grouping at the end of the matrix, and not as a sub-grouping within the matrix.

1. To do this, select the drop-down menu to the right of the **OrderYear** sub-group name  in the Grouping window, thenselect **Delete Group**.

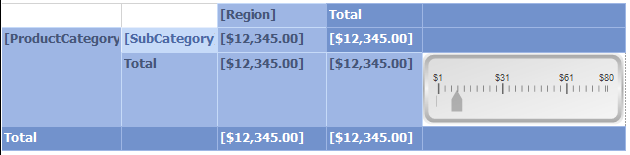


**Figure 37: Delete OrderYear group**

1. Select **Delete group and related rows and columns** in the Delete Group dialog box.

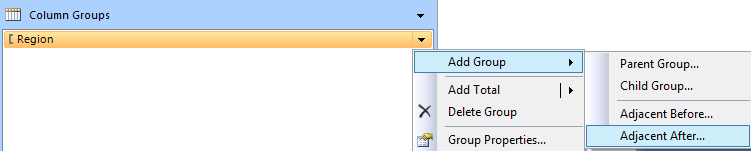


Your Tablix will now resemble the following figure.



**Figure 38: OrderYear Group Deleted**

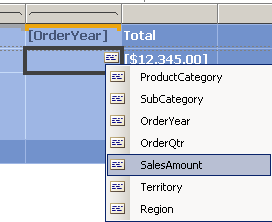
1. To add an adjacent group after **Region**, right-click the grouping window, select **Add Group**, then select **Adjacent After**.



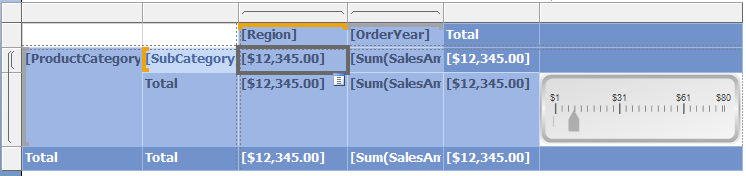
**Figure 39: Add Adjacent Group**

1. Click **OK**.
2. Select **Group By** [OrderYear] in the Tablix group dialog box.
3. Right-click in the new [OrderYear] column and add **SalesAmount**.

|  |
| --- |
| **Note** |
| SalesAmount is added to the whole column. |



Your Tablix will now resemble the following figure.



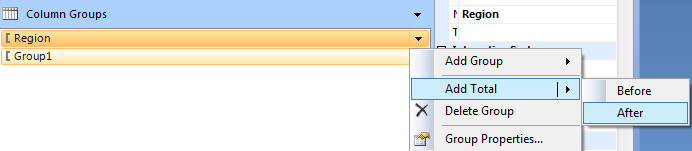
**Figure 40: New Adjacent group**

|  |
| --- |
| **Note** |
| The two group brace symbols are now alongside each other on the column headings. |

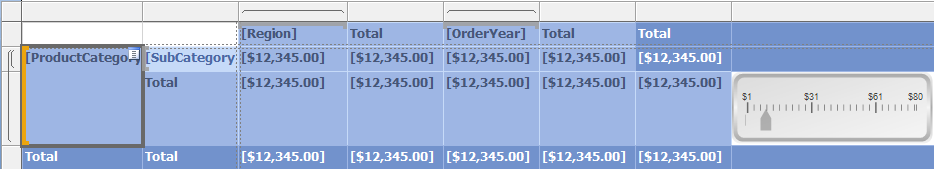
**Note**.

|  |  |  |
| --- | --- | --- |
| Changed from | **TO** |  |

1. Set all three **[Sum (SalesAmount)]** cells in the new **[OrderYear]** column to currency format by clicking each cell and using the currency button on the  Ribbon.
2. To add total columns for each of the two column groups using the grouping window, select **Add Total** in the drop-down for **Region**, then select **After**. Repeat this step for the new group.

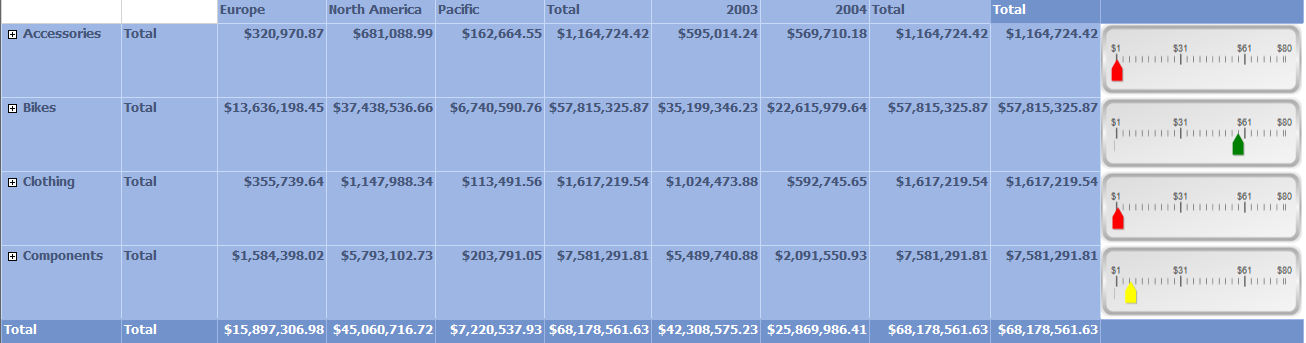


Your Tablix will now resemble the following figure.



**Figure 41: Design View of the Completed Tablix, with Parallel Dynamic Groups**

1. To run your report, click the **Home** tab and click **Run** or press **F5.**



**Figure 42: Previewing the Completed Tablix with Parallel Dynamic Groups**

* **Save your report to the server**: To save your report on the server, perform the following steps:



1. Click the Report Builder icon**, then select Save As**.
2. Select the **My Reports** folder, if present, on your local server.
3. Enter the file name **SalesbyRegion\_Ex8.rdl**.

# Exercise 9: Manual Creation of a Tablix to Match Exercise 3 (Wizard Tablix)

In this exercise, you will:

* Manually create a Tablix—similar to what you used the Matrix wizard for in Exercise 2.

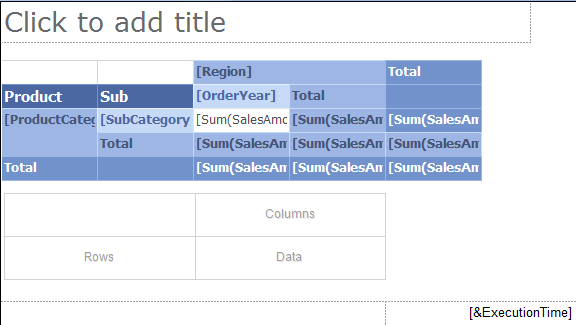
This exercise assumes that you have completed Exercise 3. You can also build this on top of later exercises, though your images will look different (with more objects). Alternatively, you can create new report and repeat Exercise 3 again.

Manually Creating a Tablix Similar to Matrix Wizard

To manually create a Tablix similar to Tablix in Matrix wizard, we use the report created in Exercise 2, **SalesbyRegion\_Ex3.rdl**. While this exercise creates a new report, using the report from Exercise 2 will enable you to compare the two versions. After opening the report, follow these steps:

1. Insert a new tablix below the existing one.
2. Create some blank space below the existing Tablix.
3. From the Insert tab of the Ribbon, select **Matrix**,then select **Insert Matrix**.
4. Resize the new matrix below the existing one.

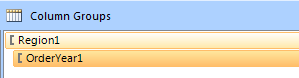
Your report in design mode will look similar to the following figure.



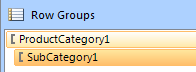
**Figure 43: Empty New Tablix**

1. From the Report Data window, do the following:
   1. Drag the **Region** field to the **Column Groups** area of the Grouping window.
   2. Drag the **OrderYear** to the **Column Groups** area of the Grouping window below **Region**.

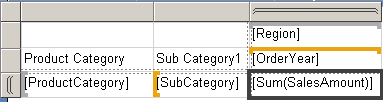
|  |
| --- |
| **Note** |
| [Orderyear] is automatically made a child of Region and is indented. |



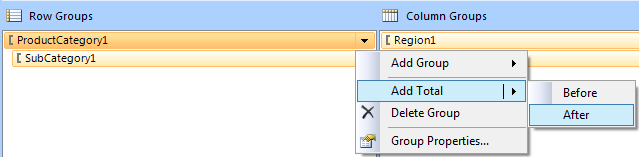
1. From the Report Data window, do the following:
   1. Drag the **ProductCategory** field to the **Row Groups** area of the Grouping window.
   2. Drag the **SubCategory** field to the **Row Groups** area of the Grouping window, below ProductCategory.

**

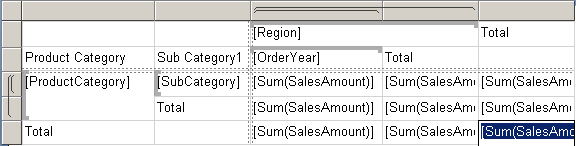
1. From the Report Data window, drag **SalesAmount** to the **Data** cell at lower right of the Tablix. The new Tablix will look similar to the following figure.



1. Toadd the totals after all of the groups in the grouping window, click each of the four groups, select **Add total**, then select **After.**



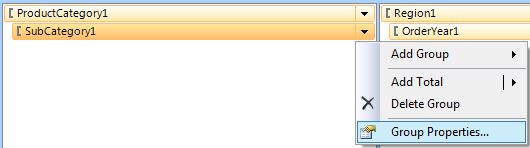
The new Tablix will resemble the following figure.



1. To run your report, click the **Home** tab and click **Run** or press **F5.**

|  |
| --- |
| **Note** |
| The data is grouped as desired, but it is difficult to initially view and understand it, as there is plenty of information. Hide the sub groups for the initial preview, just like the wizard did. Click the **Design** Tab or press F8 to switch back to Design mode. |

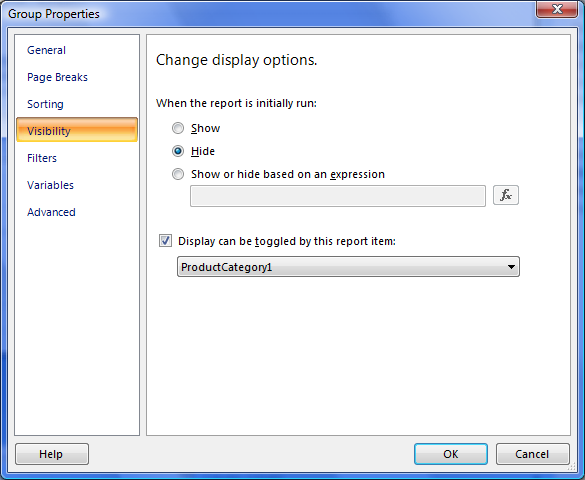
1. To hide the subgroups but allow the user to toggle them on, select **Group Properties** for **Subcategory1** from the Grouping Window.



1. In the Group Properties dialog box, select the **Visibility** category.
   1. Set **When the report is initially run** to **Hide** option.
   2. Select the **Display can be toggled by this report item** check box, then select **ProductCategory1**.

|  |
| --- |
| **Note** |
| The **1** is important as in the list where ProductCategory1 AND ProductCategory both appear, and ProductCategory is the instance from the first Tablix. |

* 1. Click **OK**.



**Figure 44: Group Properties setting the toggle item.**

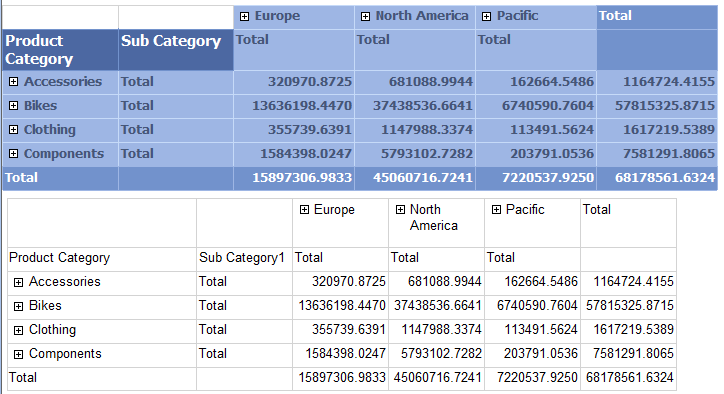
1. Repeat the same changes to the OrderYear1 subgroup:
   1. From the Grouping Window, select **Group Properties** for OrderYear1.
   2. From the Group Properties dialog, select the **‘Visibility’** page
   3. Set **When the report is initially run** to **Hide**.
   4. Select the **Display can be toggled by this report item** check box and select **Region1** from the drop-down list

|  |
| --- |
| **Note** |
| The **1** is important as in the list where Region1 AND Region both appear, and Region is the instance from the first Tablix. |

* 1. Click **OK**

1. To run your report, click the **Home** tab and click the **Run** or press **F5.**

|  |
| --- |
| **Note** |
| Other than a few formatting properties such as colors and size, the two Tablix report items are the same. |



**Figure 45: Completed Tablix.**

* **Save your report to the server**: To save your report on the server, perform the following steps:



1. Click the Report Builder icon**, then select Save As**.
2. Select the **My Reports** folder, if present, on your local server.
3. Enter the file name **SalesbyRegion\_ex8.rdl**.

# Exercise 10: Use Query, Parameter, and Filter to Limit the Tablix Data

In this exercise, you will:

* Create a new dataset that will return a unique list of the sales regions.
* Create a parameter that will use the new dataset, and will present the user the values to select from when the report is run.
* Create a filter on the tablix, which will use the value(s) the user selects from the parameter to filter/limit the data that is actually shown in the tablix after running when the report.

Filters are very useful for many scenarios such as including a single dataset for the report, but you can also represent different subsets of the data using multiple data regions (e.g. chart or tablix).

|  |  |  |
| --- | --- | --- |
| New dataset/query to build list of possible regions | Parameter uses the new dataset to present the list of possible regions to the user when they run the report | Tablix filter uses the selections made in the parameter to filer/limit the data the tablix shows. |

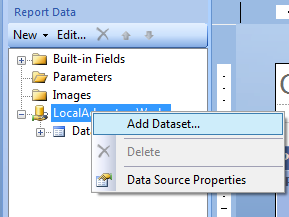
This exercise assumes that you have completed Exercise 3. You can also build this on top of later exercises, though your images will look different (with more objects). Alternatively, create new report and repeat Exercise 3 again.

Creating a New Data Set to Return a Unique List of Sales Regions

In these steps we will create a new data set, or query to a list of regions. We will then use this region list to enable users to select what regions to filter.

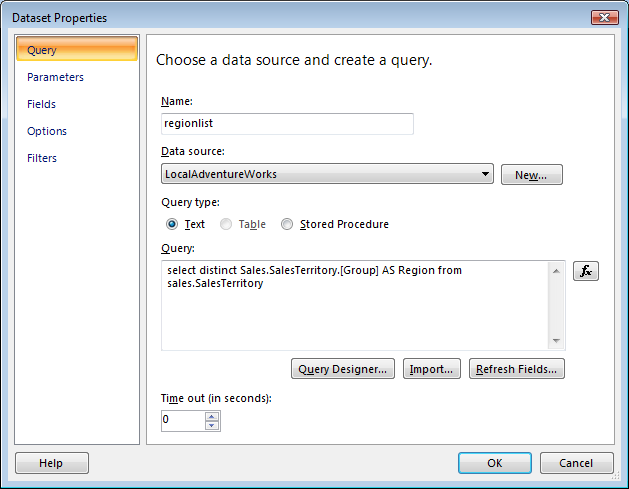
To complete this task, follow these steps:

1. Open the report “**SalesbyRegion\_Ex3.rdl**” previously created in Exercise3. If you do not have it, rerun the wizard (Exercise3).
2. In the Report Data window, right-click the Data Source and select **Add Dataset.**

**Figure 46: Adding a Dataset**

1. In the Dataset Properties dialog box, select the Query category, and do the following:
   1. Click **Add** to add a filter.
   2. Enter **Name** as **regionlist**.
   3. Set **Data Source** to **LocalAdventureWorks** (or whatever you have named your datasource)
   4. Select the **Text** option for **Query Type** and enter the following query string in the **Query** field

Select Distinct Sales.SalesTerritory.Group AS Region from sales.SalesTerritory

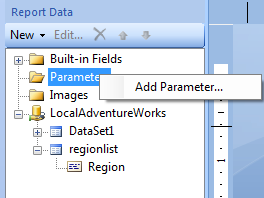
****

**Figure 47: Create New Query To Use Source The Parameter List**

1. Click **OK**

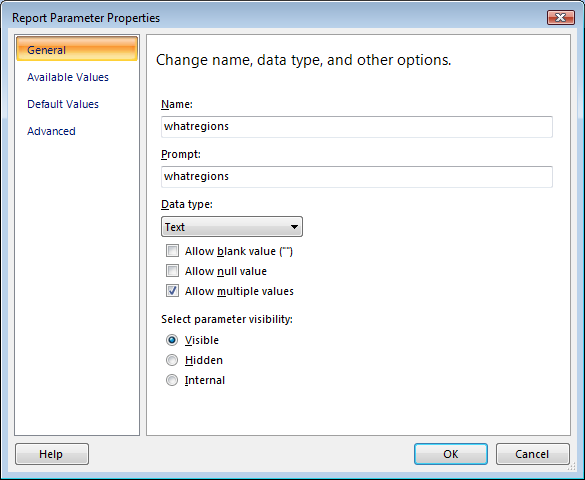
Creating a Parameter

1. In the Report Data window, right-click Parameter, then select **Add Parameter.**



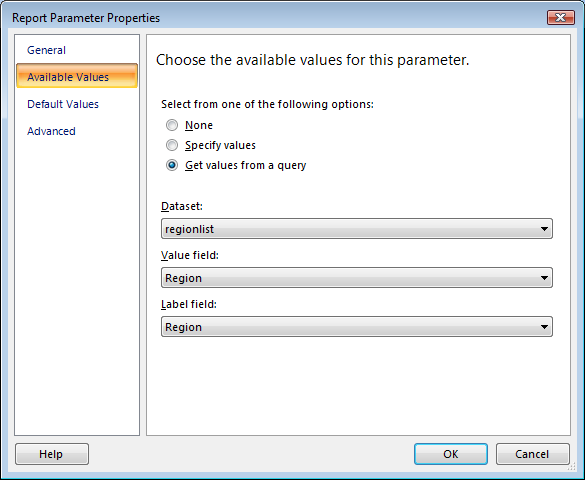
**Figure 48: Create a New Parameter**

1. In the Report Parameters dialog box, select the **General** category, then do the following:
   1. Click **Add** to add a filter.
   2. Enter **Name** as **whatregions**.
   3. Enter **Prompt** as **whatregions**.
   4. Select **Data type** as Text, then select the **Allow Multiple Values** check box.
   5. Select the **Visible** option for **Select parameter visibility**.



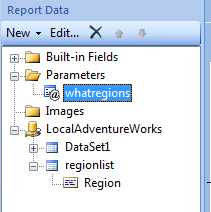
**Figure 49: Parameter General Settings**

1. Select the **Available Values** category, then do the following:
   1. Select **Get Values from query** option.
   2. Select the **Dataset** as **regionlist**.
   3. Select **Value field** as **Region**.
   4. Select **Label Field** to **Regionv**
2. Select **Allow Multiple Values** category.



**Figure 50: Parameter Available Values Settings.**

You should now see your parameter, as well as the new dataset listed in the Report Data window.

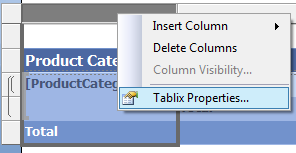


**Figure 51: New Parameter and Data Set Listed in Report Data Window**

Create the Tablix FilterThe filter in Tablix enables the user to dynamically select different region(s) to filter on each time they run the report. The user can then export the different views and, in effect, create a form of ad hoc query functionality.

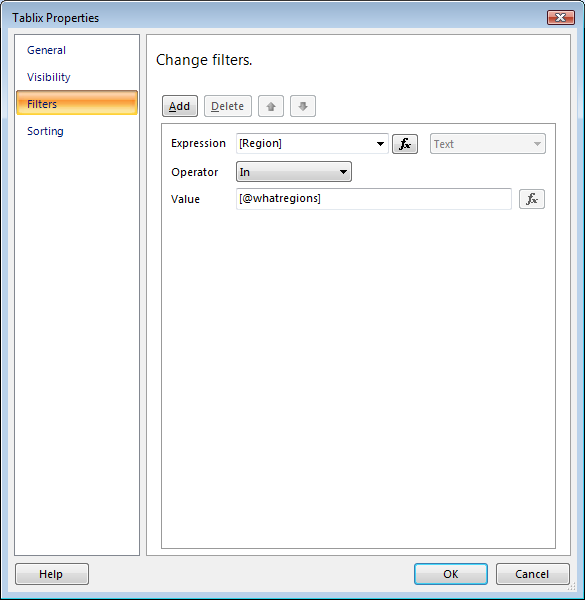
To create a filter in Tablix, perform the following steps:

1. Select the top edge of the Tablix, right-click it, and select **Tablix Properties.**

****

**Figure 52: Open Tablix Properties**

1. In the Tablix Properties dialog box, select the **Filters** category, then do the following:
   1. Click **Add** to add a filter.
   2. Set **Expression** to **[Region]** .
   3. Set **Operator** to **IN**.
   4. Set **Value** to **[@whatregions]** .

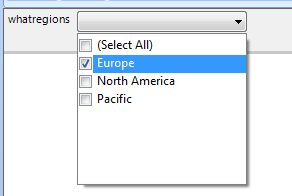


**Figure 53: Tablix Filter Settings**

1. Click **OK**.

You are now ready to run your report, and select the region from the parameter area.

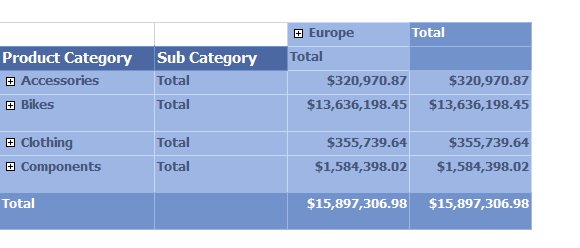
1. To run the report, click the **Home** tab, then click the **Run** button or press **F5**.
2. Select **Europe**, thenclick the **View Report** on the right.



**Figure 54: Selecting Parameter Values**

Your report will look similar to the following figure, which only shows the Europe group.

1. Switch back to the Design view.



**Figure 55: Tablix with the Filter Applied**

1. Change your filtering by simply selecting different parameters and clicking **View Report**. Repeat this a few times and view the results.

* **Save your report to the server**: To save your report on the server, perform the following steps:

1. Click the Report Builder icon **, then select Save As**.
2. Select the **My Reports** folder, if present, on your local server.
3. Enter the file name **SalesbyRegion\_ex9.rdl**.

For more information, reference the following SQL Server 2008 Online topics:[**Filter Equation Examples (Reporting Services)**](http://msdn.microsoft.com/en-us/library/cc627464.aspx)

# References and Where to Get Help

SQL Server Reporting Services home page:

<http://msdn.microsoft.com/en-us/sqlserver/default.aspx>

SQL Server 2008 Reporting Services on Microsoft TechNet:

<http://technet.microsoft.com/en-us/library/ms166352.aspx>

**SQL Server Reporting Services Forum:**

A great place to ask questions to the larger community, as well as see what others are asking and answering.

<http://forums.microsoft.com/Forums/ShowForum.aspx?ForumID=82&SiteID=1>

**SQL Server 2008 Reporting Services Books Online (Help):**

<http://msdn.microsoft.com/en-us/library/ms159106(SQL.100).aspx>

**www.codeplex.com for Adventureworks sample database and report downloads:**

<http://www.codeplex.com/MSFTDBProdSamples/Release/ProjectReleases.aspx?Released=18407>

**Main SQL Server samples page:**

<http://codeplex.com/SqlServerSamples>

**Microsoft Connect Feedback site:**

<http://connect.microsoft.com>

A connection point between you and Microsoft, and ultimately the larger user community. Your feedback enables Microsoft to make software and services the best that they can be, and you can learn about and contribute to exciting projects.