## Working with SQL Reporting Services

**Lab Time**: 60 minutes

**Lab Folder**: c:\Student\Modules\SSRS\Labs

**Lab Overview**: In this lab you will work with SQL Server Reporting Services to create a SQLReport Builder report, connect it to the WingtipSales database, add parameters, and publish the report to SharePoint.

### Exercise 1: Create Report Library and Report Data Sources

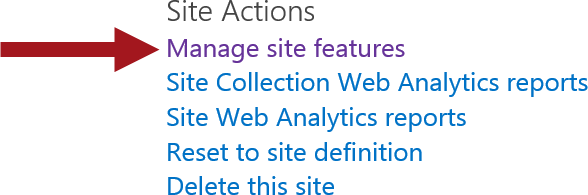
In this exercise you will work within a BI Center site to create a Report library and a Shared Data Source for SQL Reporting Services in SharePoint 2013.

#### Log in to Student Environment

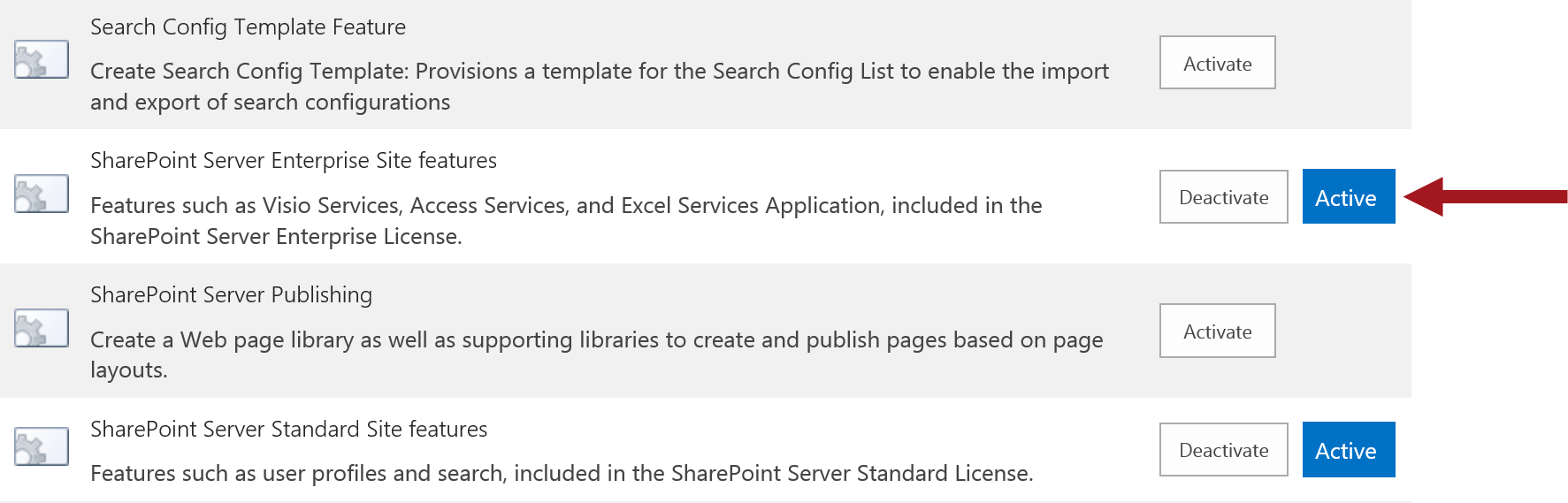
1. Login to the Student VM using the login **WINGTIP\Administrator** and the appropriate password.
   1. If you’re using a local VM provided by the hosting training company, the password will be **Password1**.
   2. If your student VM is hosted by CloudShare, the password for the **WINGTIP\Administrator** account is going to be unique for each student, system-generated by CloudShare. Also note that the CloudShare VM configuration usually logs you into the VM automatically so you do not have to enter the user name and password.

#### Enable Features in SharePoint Site

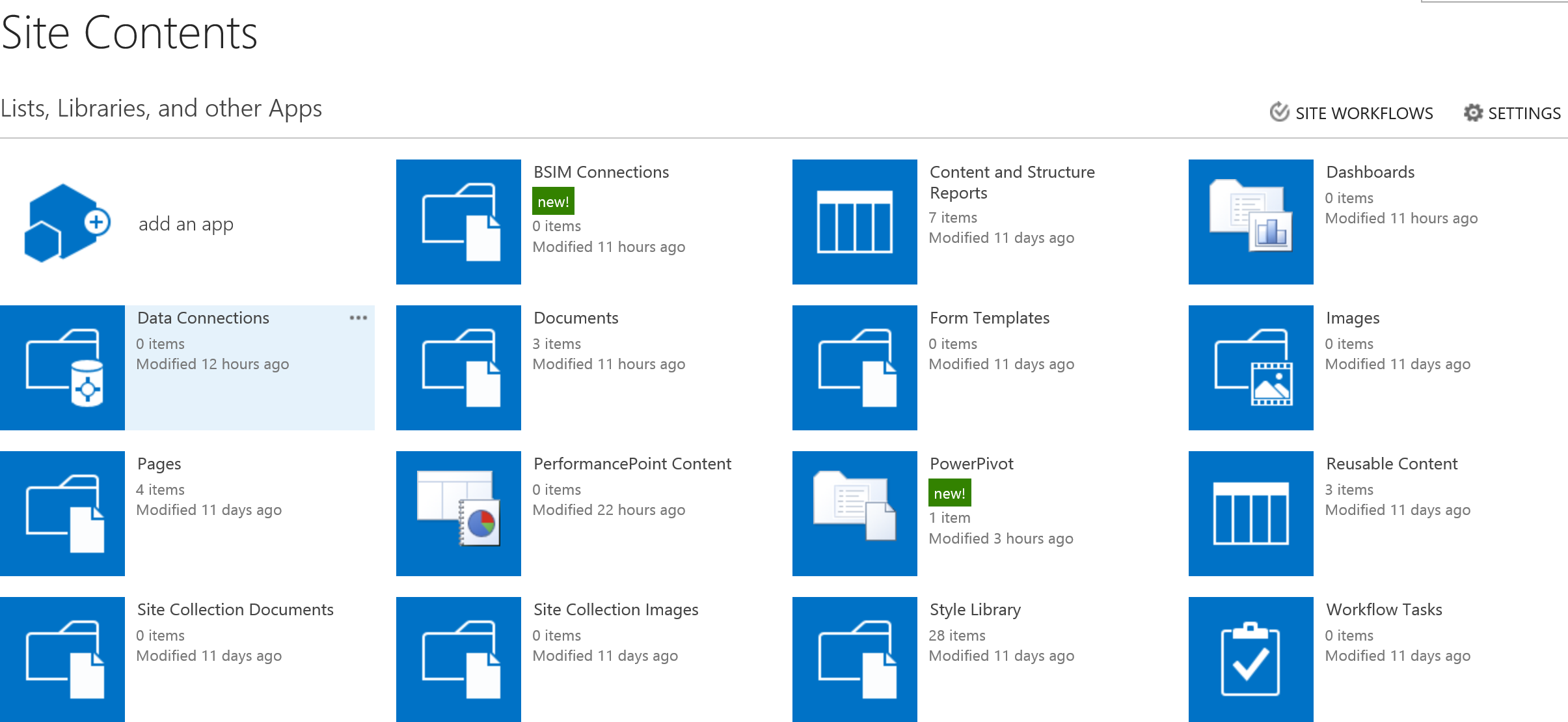
1. Open the Business Intelligence Center site: [**http://intranet.wingtip.com/sites/bicenter**](http://intranet.wingtip.com/sites/bicenter)
2. Enable the site-level feature with the title of **SharePoint Server Enterprise Site features** to make the **Report Library** template available for creating new document libraries with the current site.
   1. Use the **Site Actions** menu to navigate to the **Site Settings** page.
   2. On the **Site Settings** page, locate and click the link in the **Site Actions** section with the caption **Manage site features**.



* 1. On the **Site Features** page, activate the site-level feature with the title **SharePoint Server Enterprise Site features**.



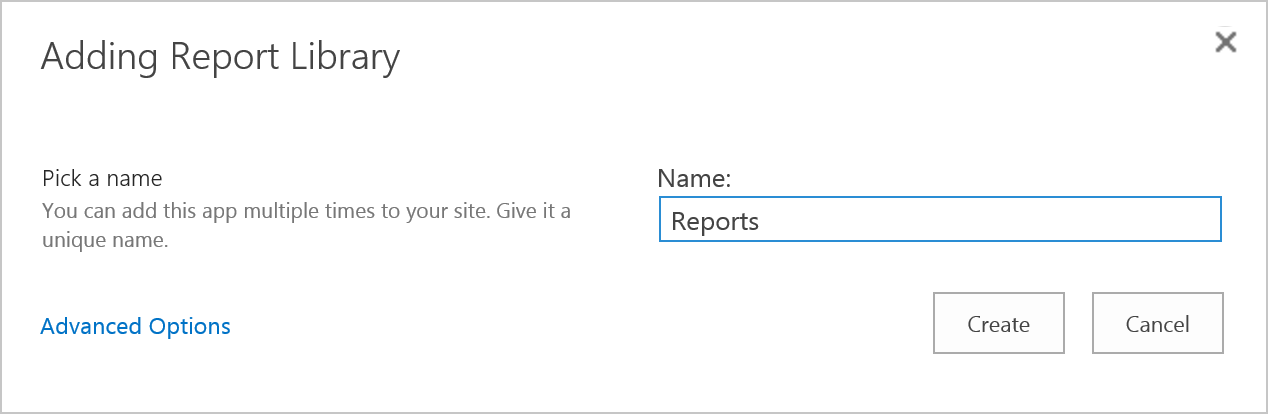
1. Create a new **Reports Library** with a title of **Reports**.
   1. Use the Site Actions menu to navigate to the **Site Contents** page.
   2. On the **Site Contents** page, you should be able to see that there are a few existing document libraries in the site such as **Dashboards** and **Data Connections**. However, there is no document library for reports that have been created with SQL Server Reporting Services.



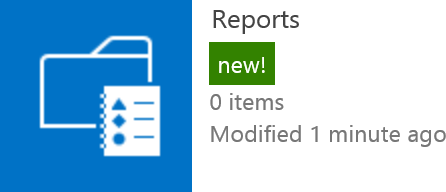
* 1. On the **Site Contents** page, click **add an app** to navigate to the **Your Apps** page.
  2. On the **Your Apps** page, click on the **Report Library** tile to begin the process of creating a new report library.



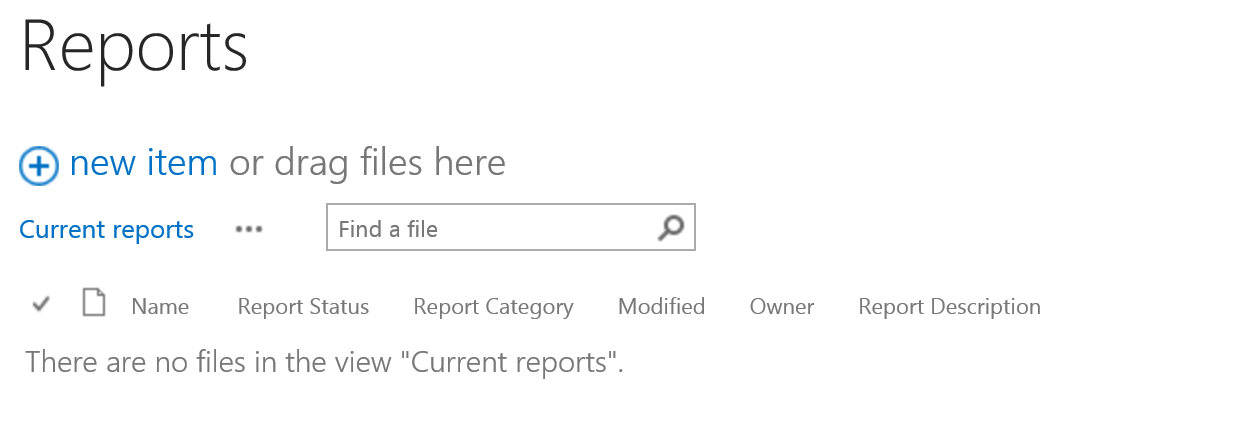
* 1. On the Adding Report Library page, type **Reports** in the **Name** text box and click **Create**.



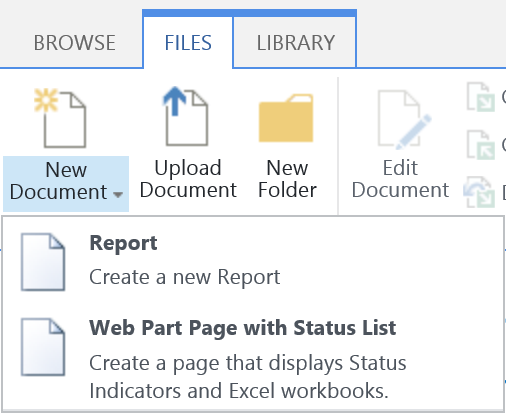
* 1. Once you have created the **Reports** library, you should be able to see a tile for it on the **Site Contents** page.



* 1. Click on the **Reports** tile to navigate to the default view of the Reports library.

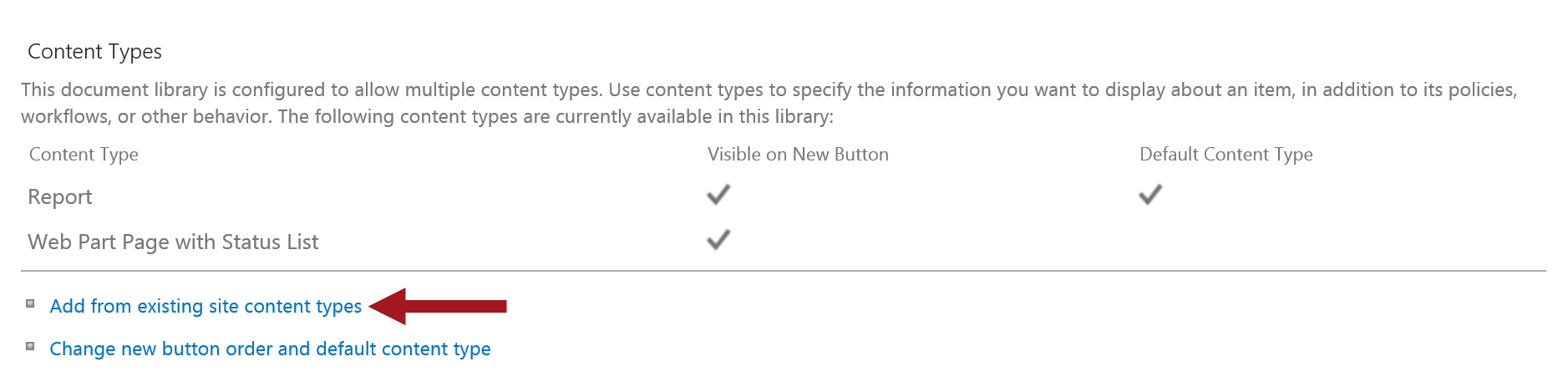


* 1. From the **FILES** tab, click the **New Document** ribbon button drop-down menu. You should notice two content types are available: **Report** and **Web Part Page with Status List**.

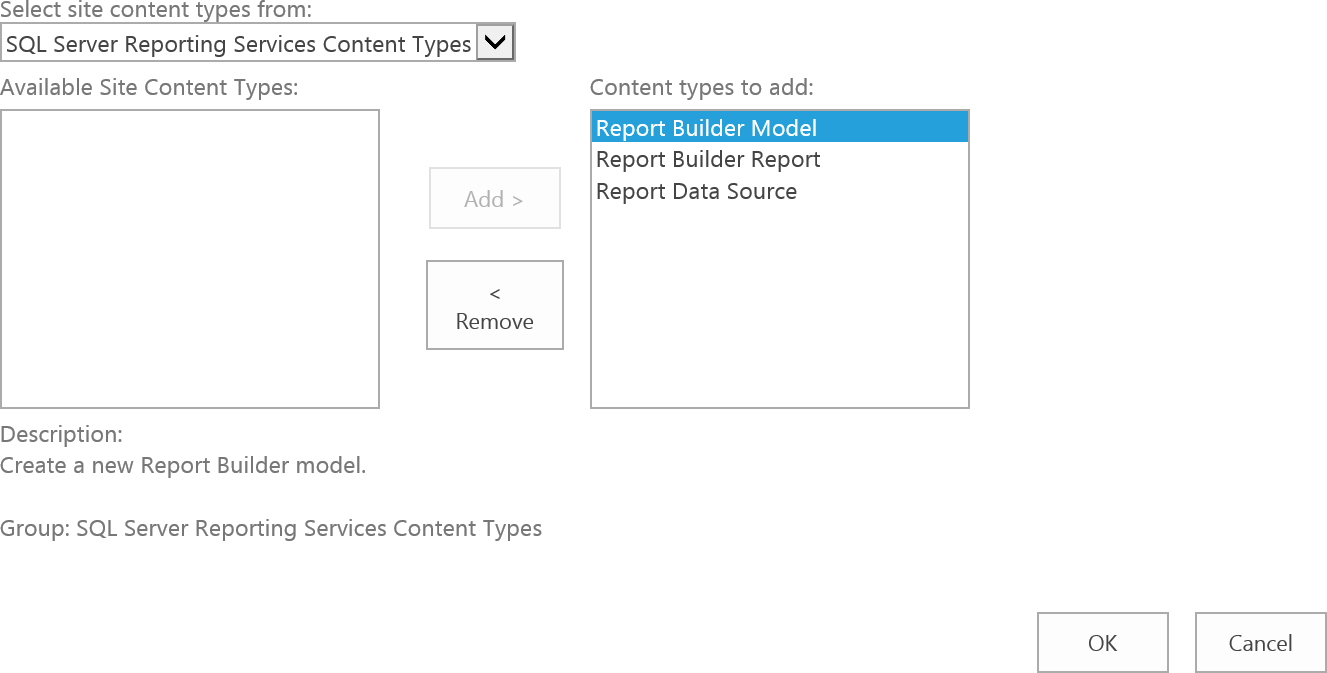


Even though the Report library has been created, it only contains two content types which are **Report** and **Web Part Page with Status List**. However, you must configure the Reports library to support a few additional content types so business users can create new documents of the proper type.

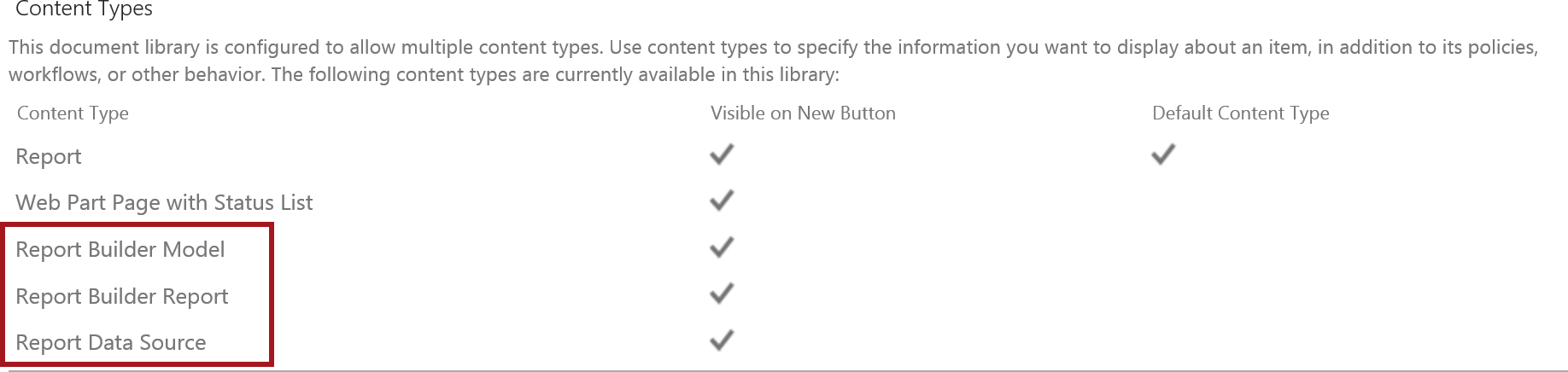
1. Configure the **Reports** Library to support additional content types.
   1. From the **LIBRARY** tab in the ribbon, click on the **Library Settings** ribbon button.
   2. Under the Content Types section, click the **Add from existing content types** link.



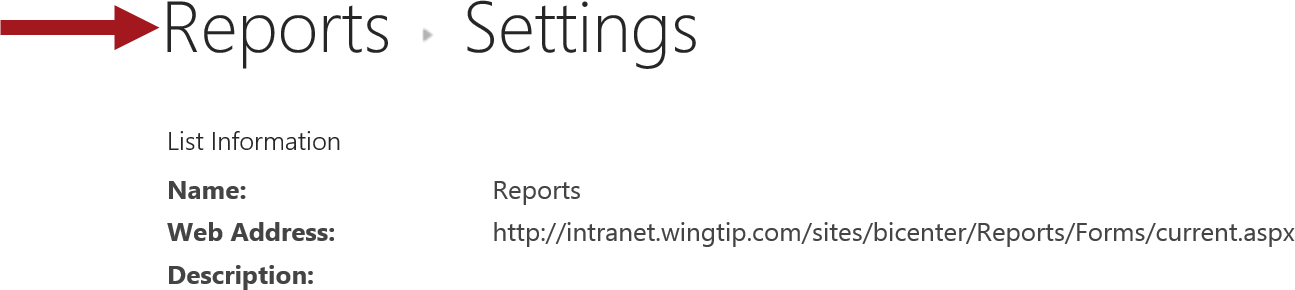
* 1. From the **Select site content types from** drop-down, select **SQL Server Reporting Services Content Types**.
  2. Add the **Report Builder Model**, **Report Builder Report** and **Report Data Source** content types and then click **OK**.



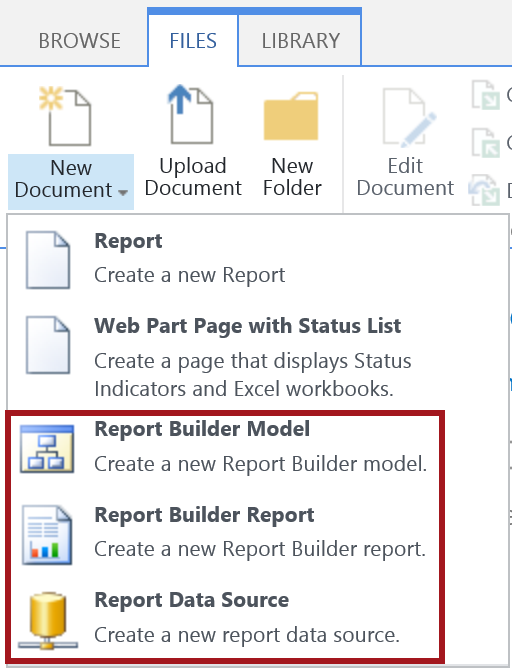
* 1. Notice the newly added content types are displayed in the Content Types section.



* 1. Click the **Reports** link in the title navigation to go back to the library default view.

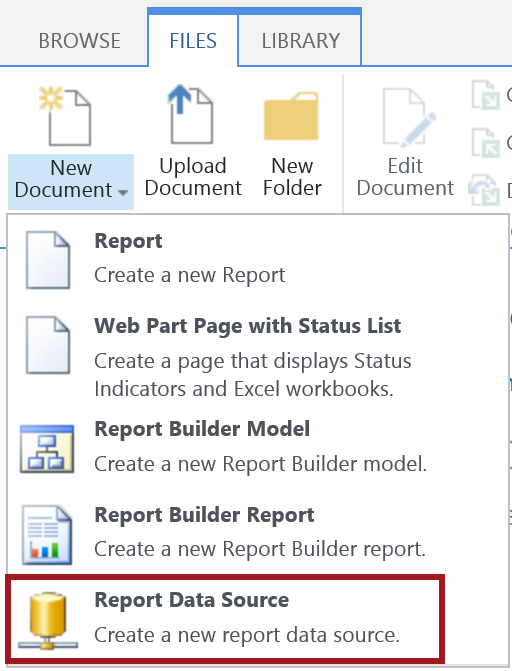


* 1. From the **FILES** tab, click the **New Document** ribbon button drop-down and notice the additional content types are now available.



#### Create Data Source for Wingtip Sales

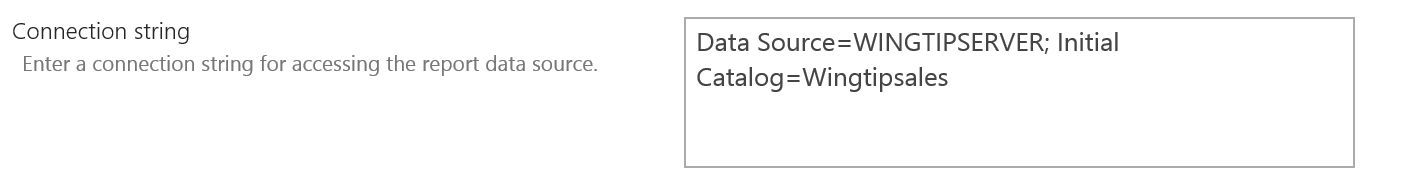
1. Create a SQL Server **Report Data Source** for the **WingtipSales** database in SQL Server 2012.
   1. Navigate to the default view of the **Reports** library and activate the **FILES** tab. Use the **New Documents** dropdown menu and click on new **Report Data Source**.



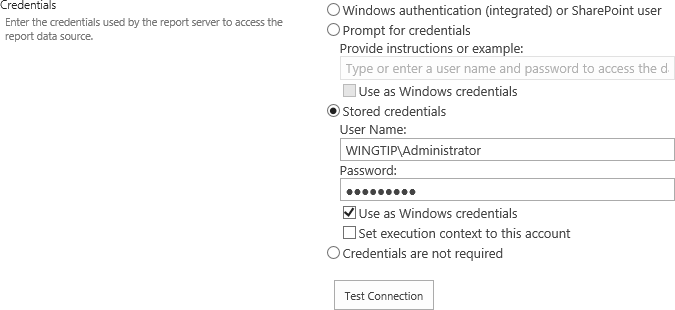
* 1. In the **Data Source Properties** page enter a Name of **WingtipSalesConnection**.
  2. In the **Data Source Type** drop-down, select **Microsoft SQL Server**.



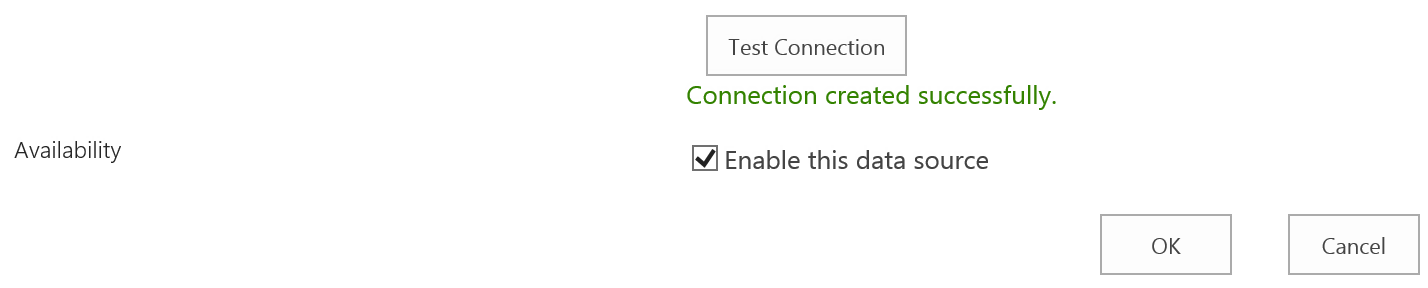
* 1. In the Connection String text box, type **Data Source=WINGTIPSERVER; Initial Catalog=WingtipSales**.



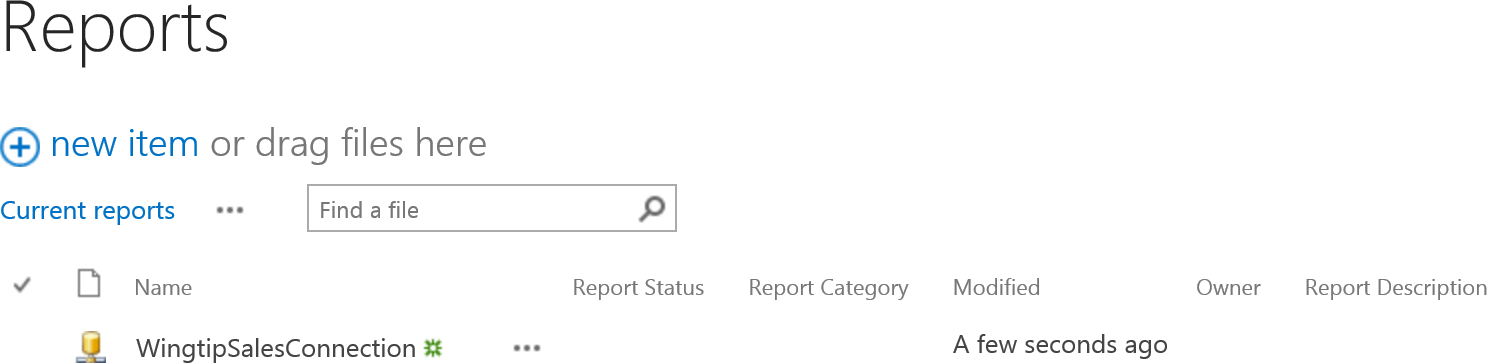
* 1. In the **Credentials** section, select the **Stored Credentials option** and set the **User Name** to **WINGTIP\Administrator** and the **Password** to **Password1**. Also check the box for **Use as Windows credentials** and then.



* 1. Click the **Test Connection** button.



* 1. If the message **Connection created successfully** is displayed, click **Enable this data source** and click **OK** to finish creating the new data connection.

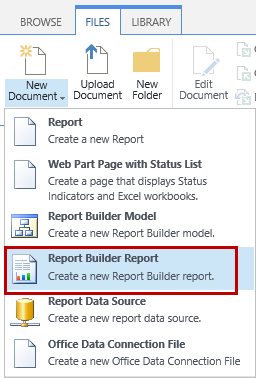


### Exercise 2: Create Wingtip Sales Map Report

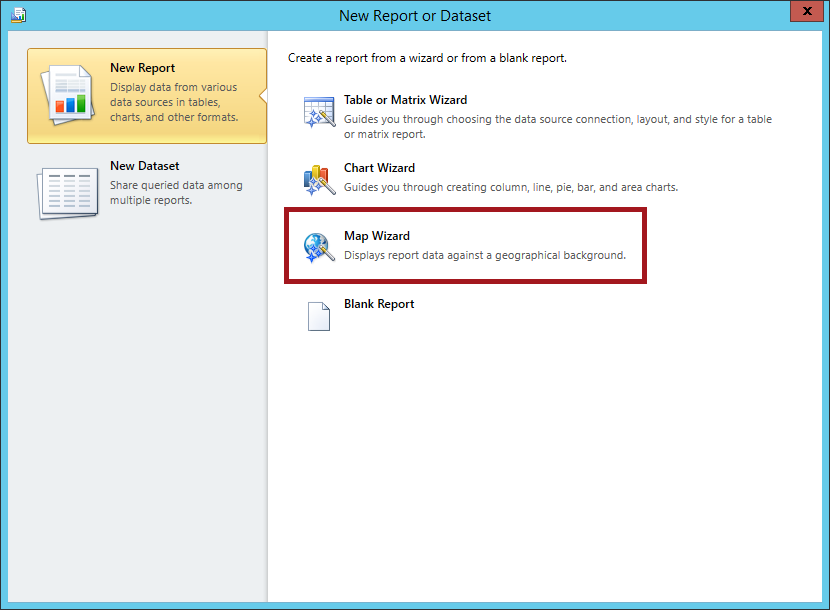
In this exercise you will create a SQL Server Reporting Services Map report using the WingtipSalesConnection data connection you created in the previous exercise.

#### Create Report Builder Report

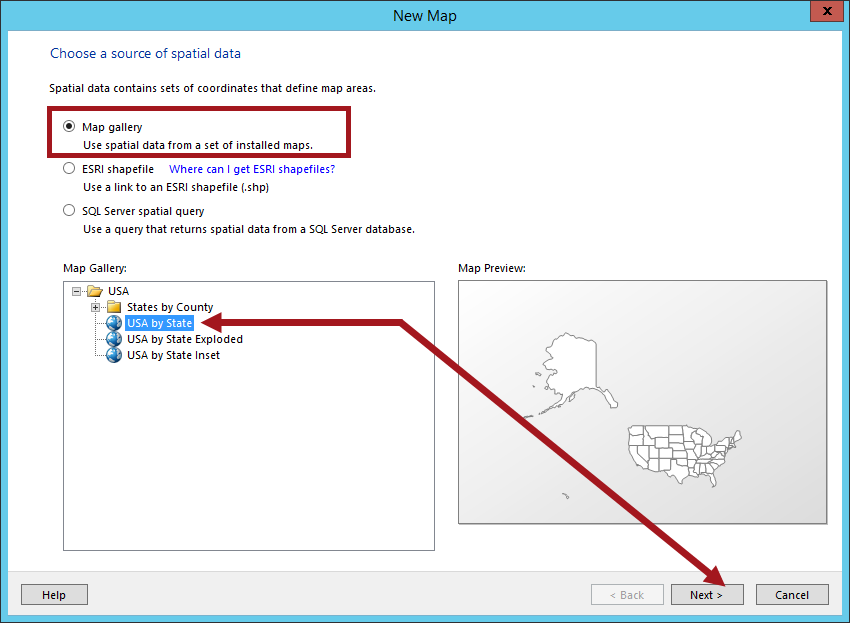
1. Create a Map Report by following the steps below:
   1. Ensure the BI Center site is open <http://intranet.wingtip.com/sites/BICenter>
   2. In the Reports library from the **New Documents** menu, click **Report Builder Report**. This action will load the Report Builder application.



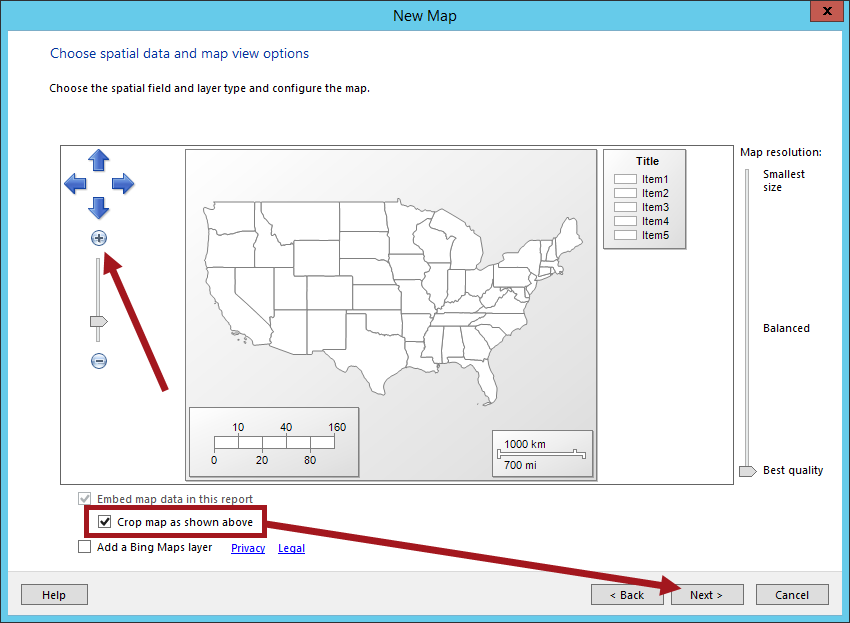
* 1. Once the Report Builder application has started up, click on **Map Wizard**.



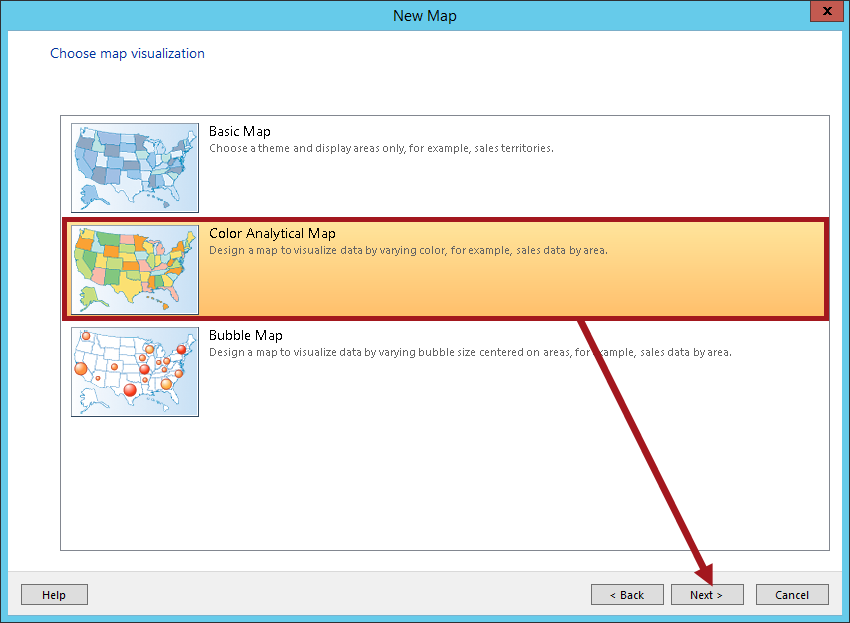
* 1. In the **Choose a source of spatial data** dialog, ensure the **Map gallery** option is selected. Select **USA by State** and then click **Next**.



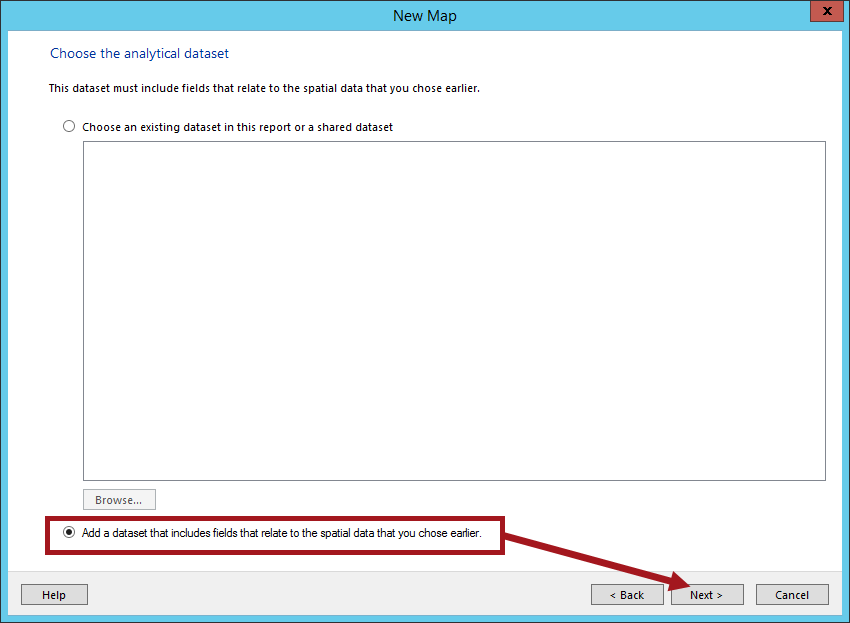
* 1. In the **Choose** **spatial data and map view options** dialog, resize the map so the US is in the view. Check **Crop map as shown above** and then click **Next**.



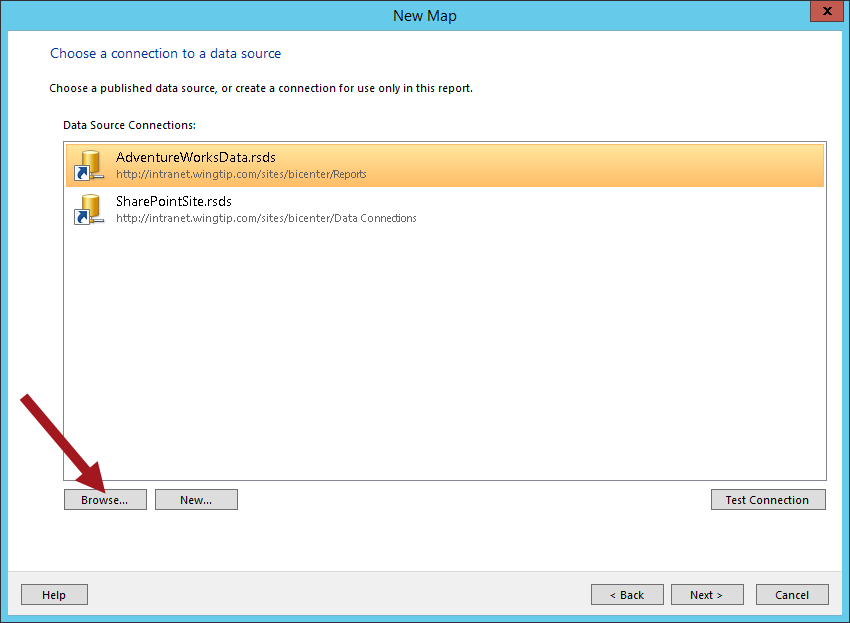
* 1. In the **Choose map visualization** dialog, select **Color Analytical Map** and then click **Next**.



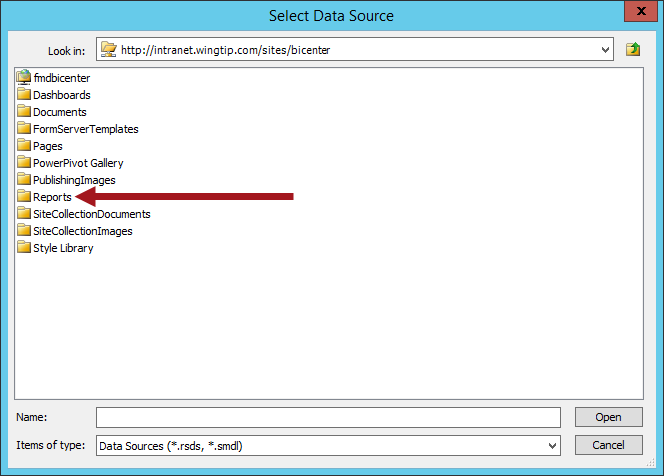
* 1. In the **Choose the analytical dataset** dialog, select **Add a dataset that includes fields that relate to the spatial data that you chose earlier** and click **Next**.



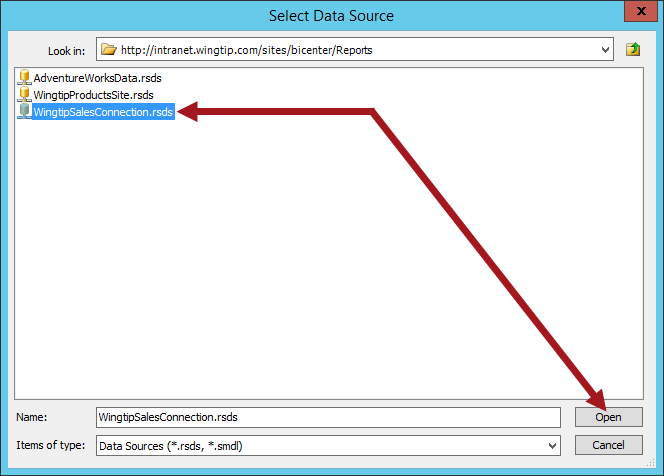
* 1. In the **Choose a connection to a data source** dialog, Click **Browse**.



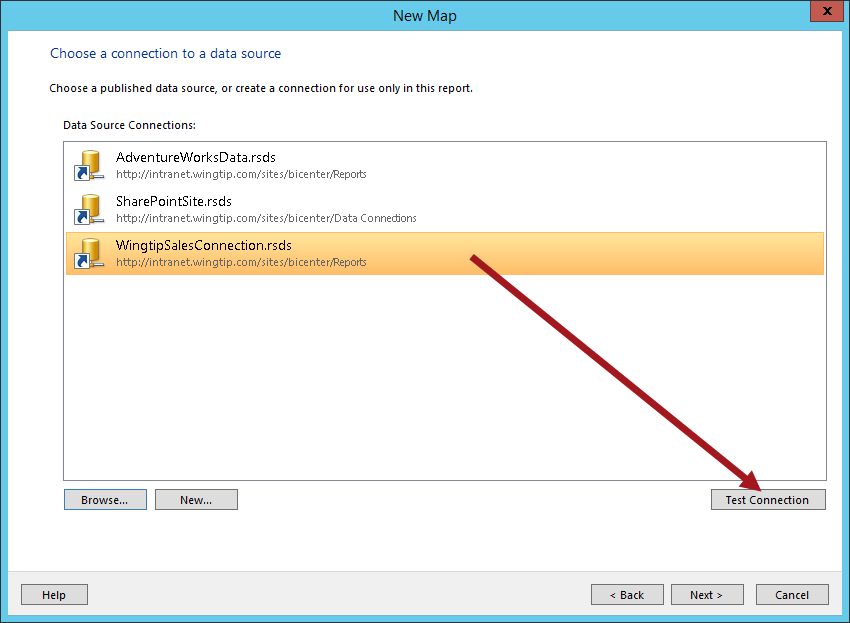
* 1. Double click on **Reports**.



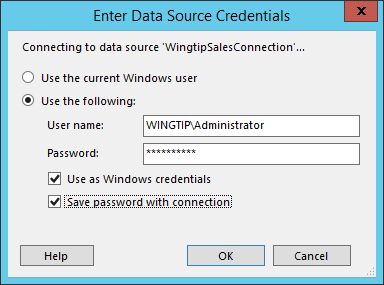
* 1. Choose **WingtipSalesConnection.rsds** and click **Open**.



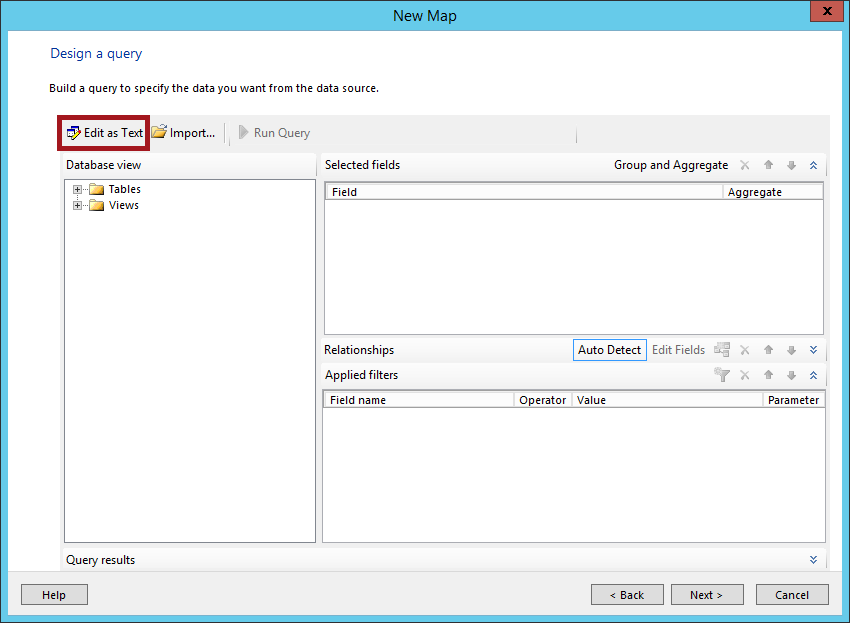
* 1. The **WingtipSalesConnection.rsds** should now be selected in the dialog by default. Click **Test Connection**.



* 1. If the connection is successful click **OK** then click **Next** to move to the **Design a query** page.
  2. If prompted for credentials, enter the credentials for the **WINGTIP\Administrator**. Select **Save password with connection** and then click **OK**.



* 1. Click **Next** and then the **Design a query** dialog will display. Click on **Edit as Text** so that you can enter the raw text for a SQL Server query.

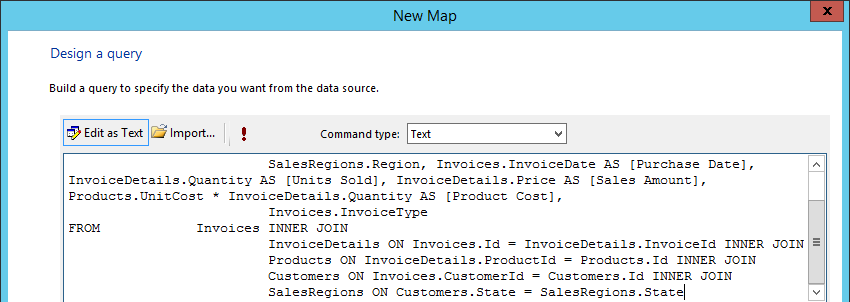


There are two ways you can build the query using the query designer in Report Builder. You can use the design builder by expanding and clicking on the tables and columns you want to query or use the Edit as Text option to use Transact-SQL (T-SQL). For this lab we will use T-SQL to build our query. Also, to make things a bit easier we have provided a text file in the folder for this lab which has the text for the SQL query that you will use. You can simply copy-and-paste the text for the required SQL query into the Report Builder application.

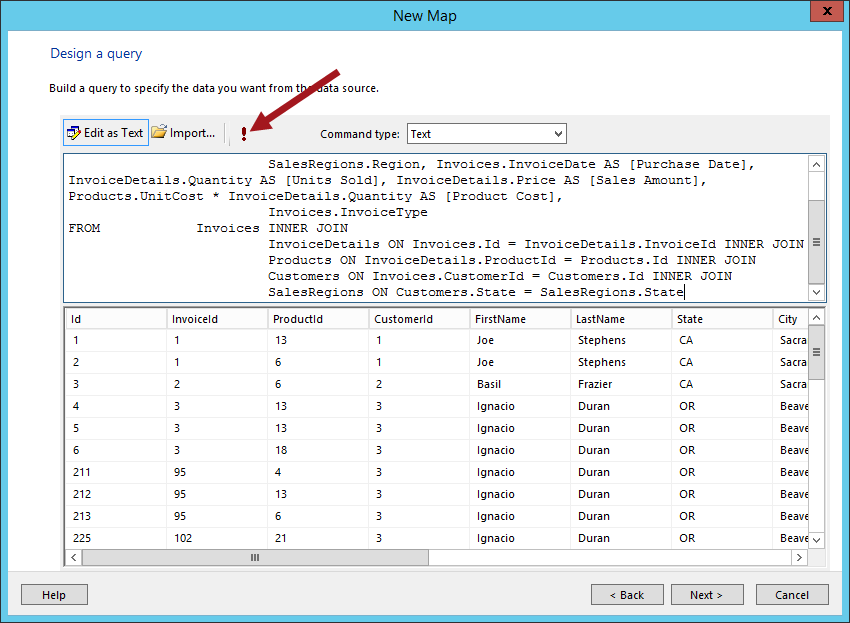
* 1. Locate the text file named **WingtipSalesMapSQLQuery.txt** located at the following path.

C:\Student\Modules\Reporting\Lab\WingtipSalesMapSQLQuery.txt

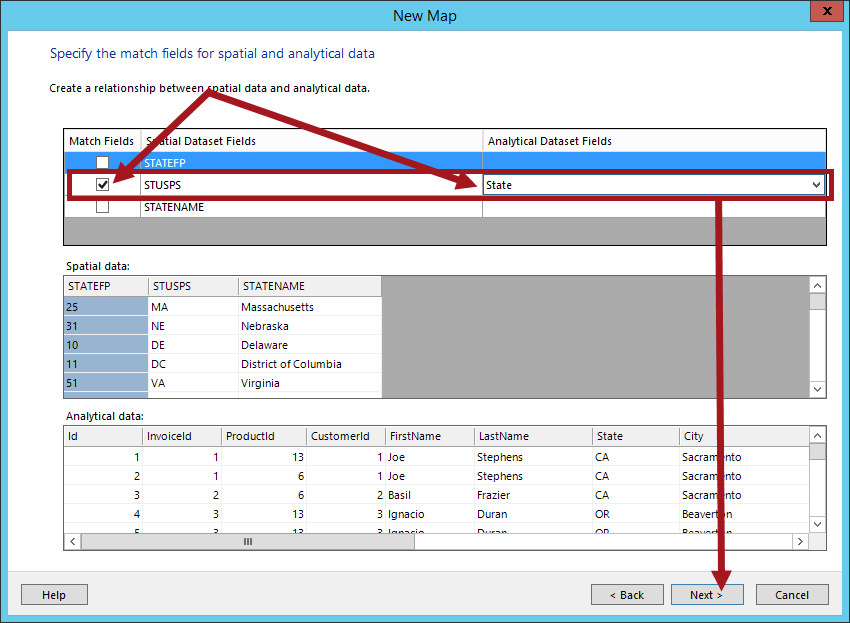
* 1. Open **WingtipSalesMapSQLQuery.txt** using the Windows utility program **NOTEPAD.EXE**.
  2. Select all the text from inside **WingtipSalesMapSQLQuery.txt** file and copy it to the Windows Clipboard
  3. Return back to the **Design a query** page in the **Report Builder** application. Paste the text with the SQL query from the Windows Clipboard into the multiline text box directly under the **Edit as Text** button.



* 1. Click the **exclamation icon (!)** to test the query and then click **Next**.

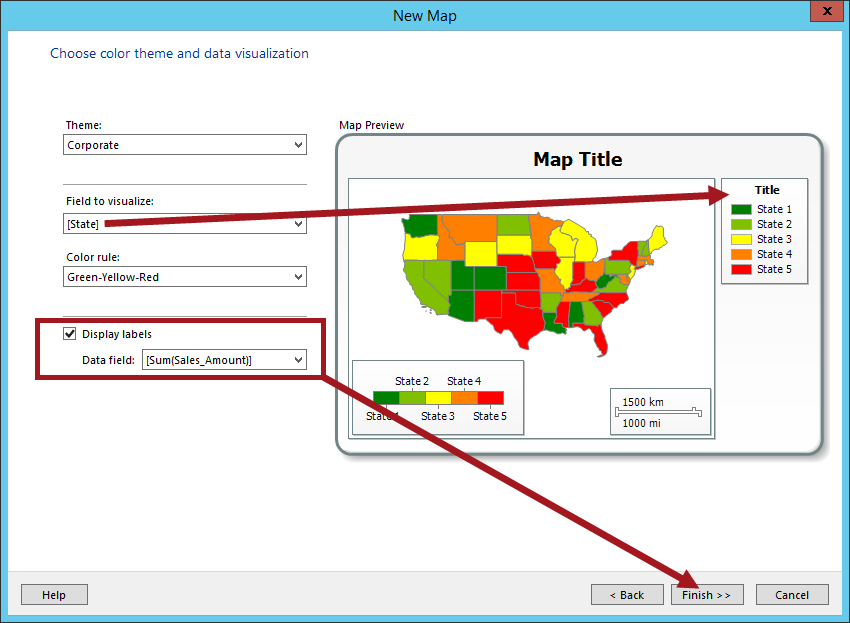


* 1. The next page in the wizard is that **Match fields for spatial and analytical data** page which will allow you match the spatial dataset fields to the analytical dataset fields.
     1. Check the **Match Fields** option for **STUSPS**.
     2. In the **Analytical Dataset Fields** drop-down list for **STUSPS**, choose **State** then click **Next**.



Since we will use the default Spatial data for the STATEFP and STATENAME, the only field that needs to be mapped is State.

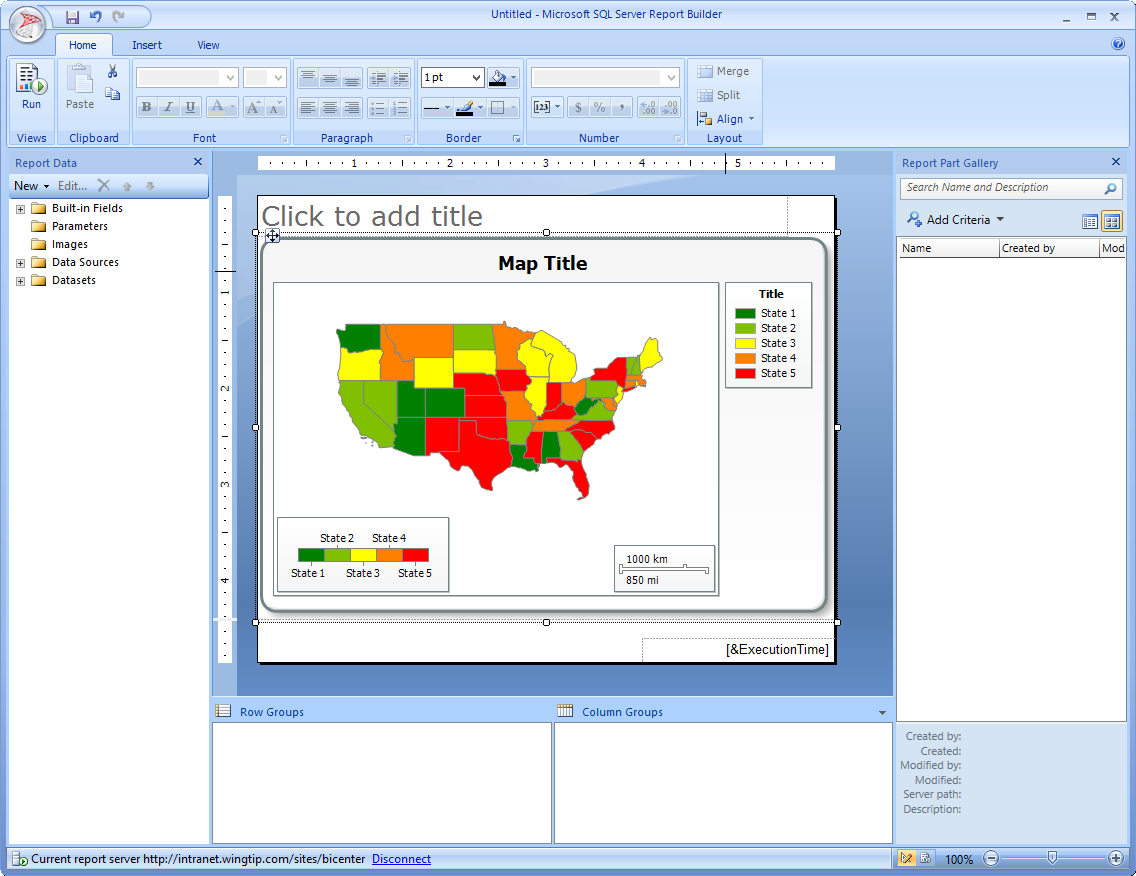
* 1. From the **Choose color theme and data visualization** dialog, select the following options:
     1. Set the **Theme** to **Corporate** (or any other desired theme).
     2. Set the **Field to visualize** to **[State]**.
     3. Set the **Color rule** to **Green-Yellow-Red** (or any other desired color rule).
     4. Check the option for **Display labels** and then in the data field drop-down list choose **[Sum(Sales\_Amount)]**.
  2. Your settings should now look similar to the figure below.



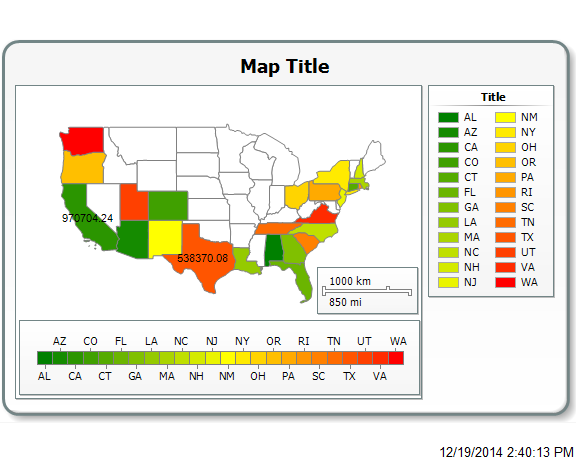
If you set the Field to visualize to #STUSPSthis would display all states even if they do not contain data however we chose [State] instead so the map will only color the states that have data. If the data is updated and new states contain data, the map would automatically color the new states on the map that have data.

* 1. Click **Finish**.

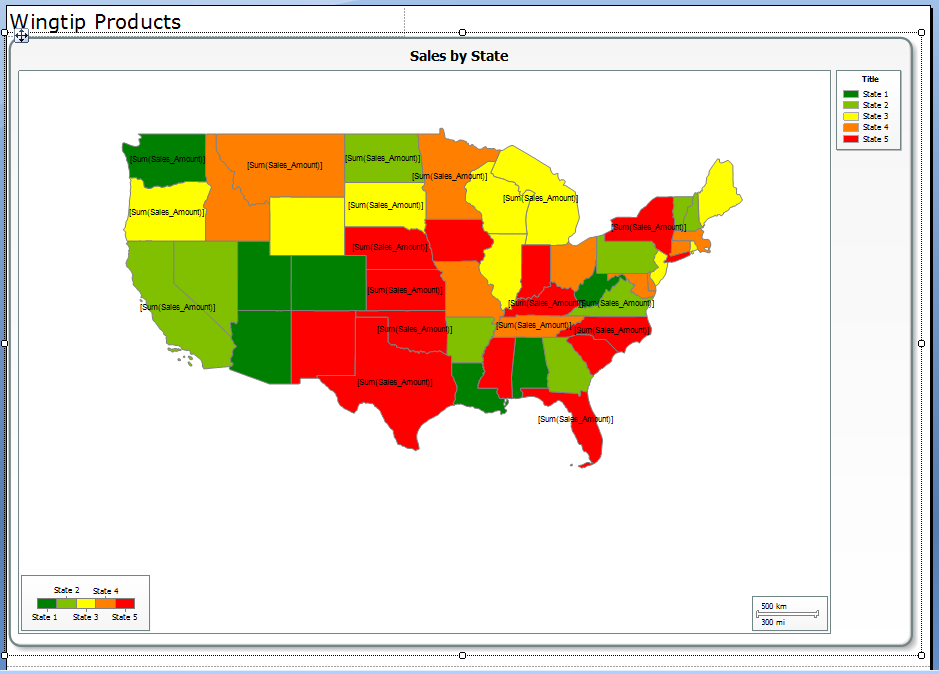
1. At this point, you have completed using the wizard to create a new map report. You should now see the new report you just created in the report designer of the Report Builder application.



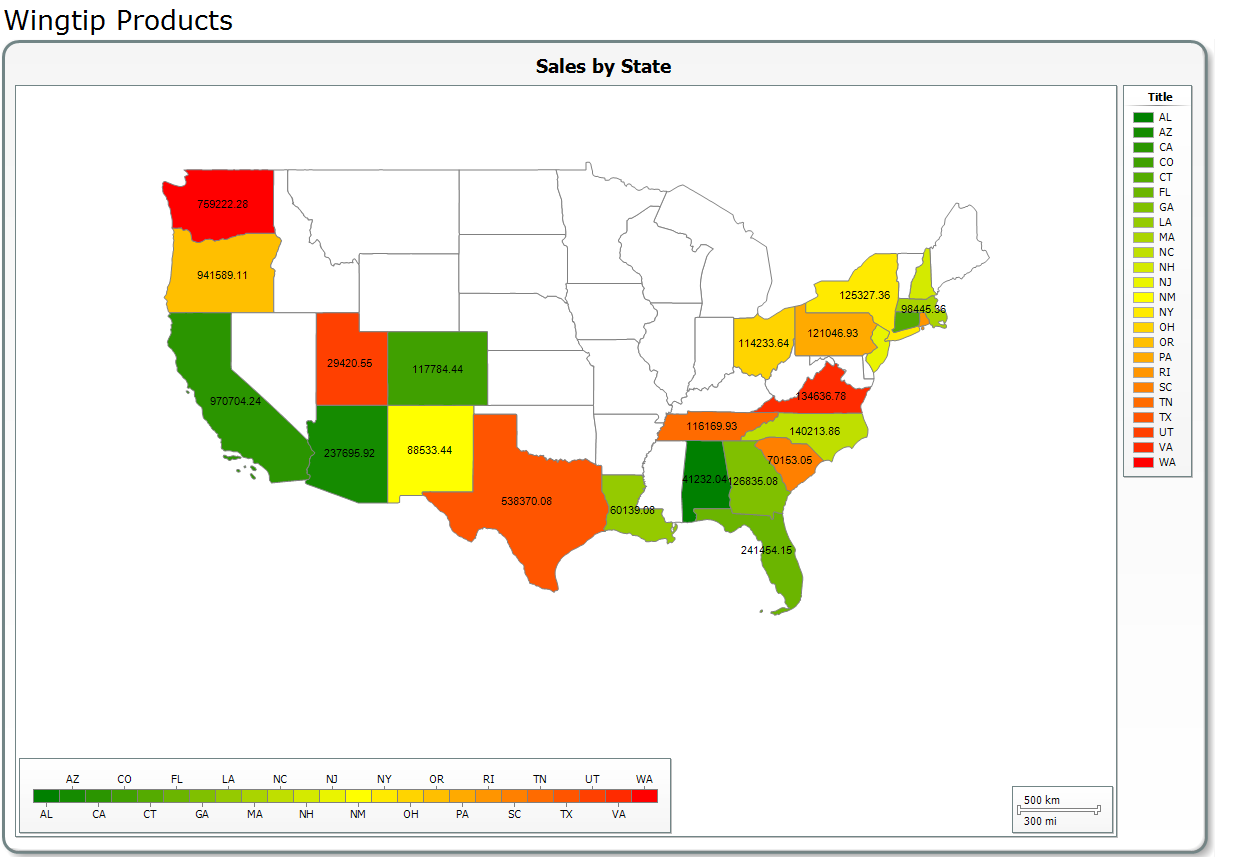
* 1. Run the report to see what it looks like. You should be able to see some of the data however the map will be too small to show all data therefore you will need to make modifications to the map.



* 1. Return to design view. Now add a title to the report by clicking in the **Click to add title** type **Wingtip Products**.
  2. Change the **Map Title** to **Sales by State** by double-clicking on Map Title and then replacing the text.
  3. Adjust the report canvas and the map to a larger, desired size until you see the data on the map.

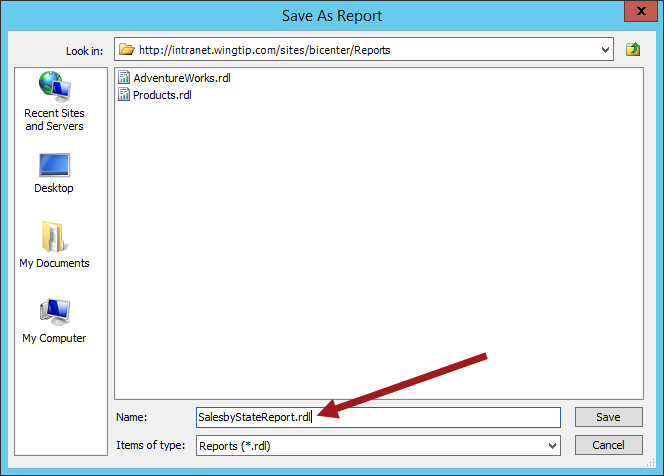


* 1. Run the report to test.

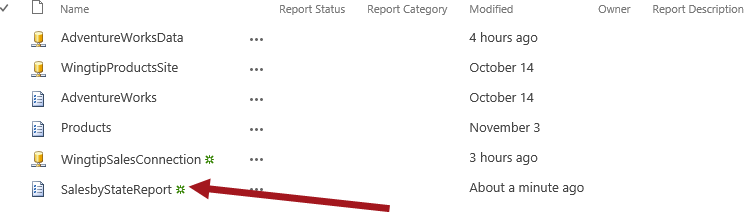


#### Save and Test Report in Browser

1. Click the **Save** icon in the ribbon to save the report. When you save the report, save it into the **Reports** library in the BI Center site using the file name **SalesbyStateReport.rdl**.

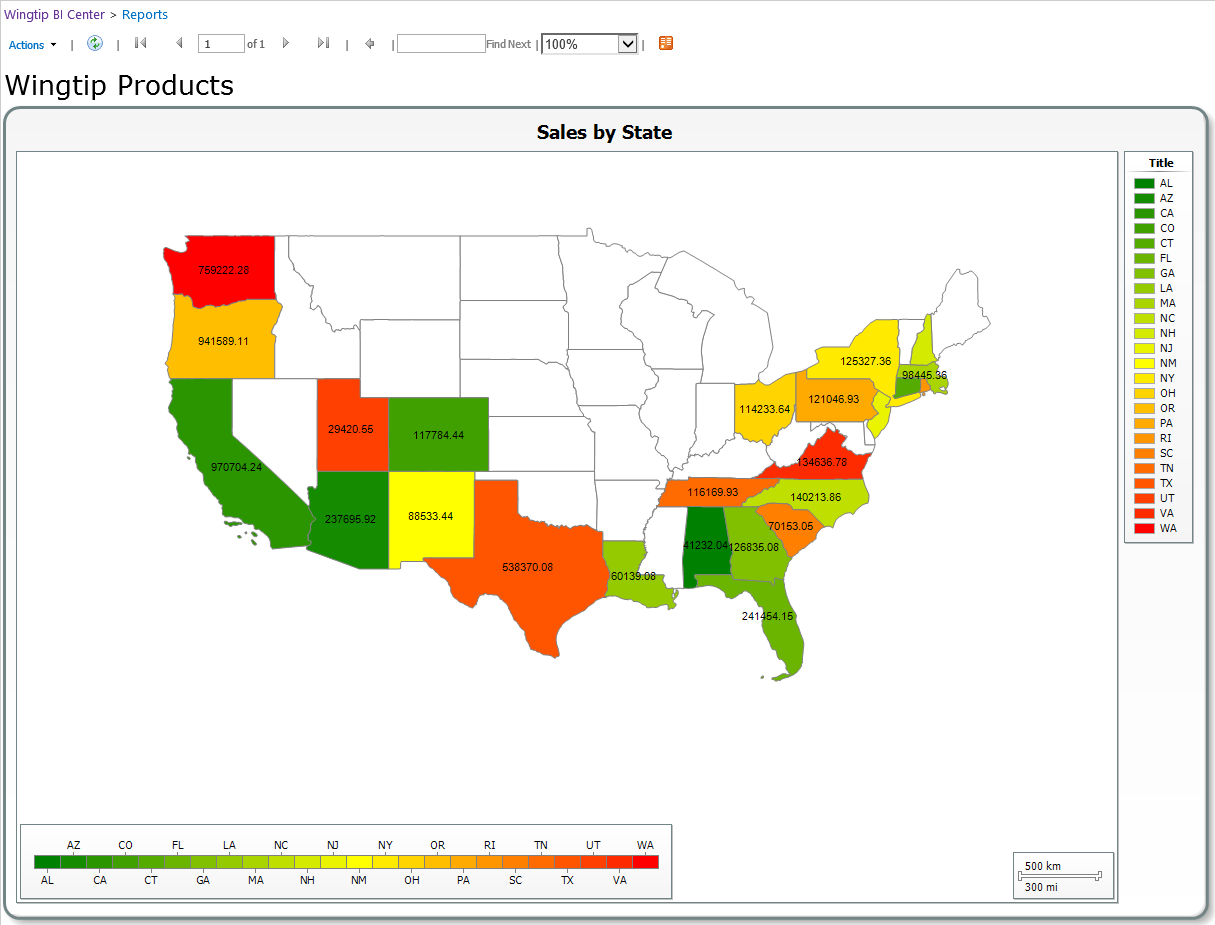


* 1. Confirm that the new report you just created named **SalesbyStateReport** can be seen in the default view of the **Reports** library.



Even though we will be making more changes to the report it is best practice to save the file incase the Report Builder or computer crashes. This ensures you do not lose any work that has been completed on the report.

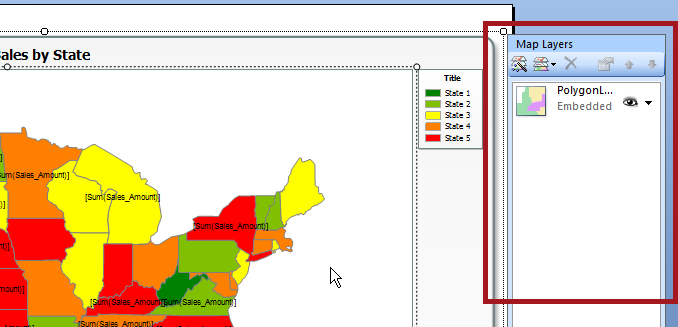
1. Now test the **SalesbyStateReport** report in the browser.
   1. Click **SalesbyStateReport**.
   2. Confirm that the report displays properly inside the browser.



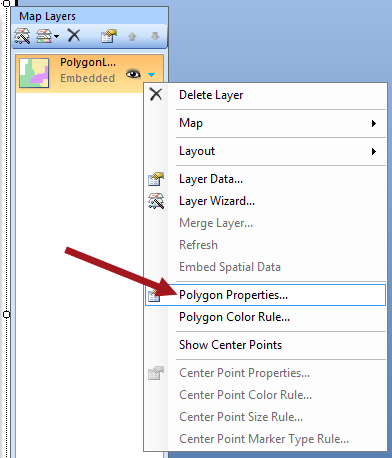
Notice the values are not showing in currency format. Proceed to the next steps to update the map in Report Builder to change the format of the value.

#### Modify Report Properties

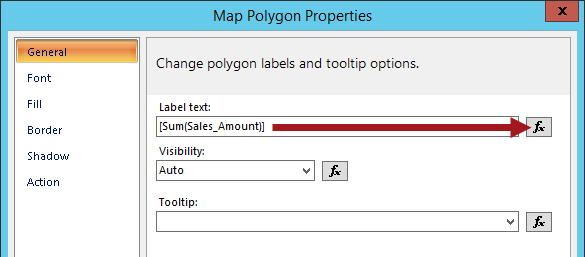
1. To update the Map field value follow the steps below:
   1. Go back to the report in Report Builder.
   2. Click outside of the US in the whiter area of the map to activate the Map Layers panel.



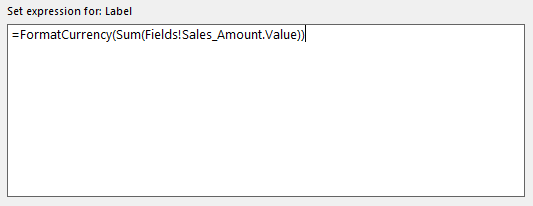
* 1. In the **Map Layers** panel, click the drop-down arrow for the **PolygonLayer1** and select **Polygon Properties…**



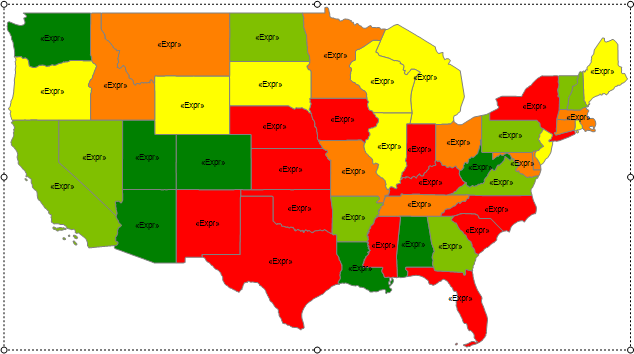
* 1. Click the function button (**fx**) for **the [Sum(Sales\_Amount)]** **Label text**.



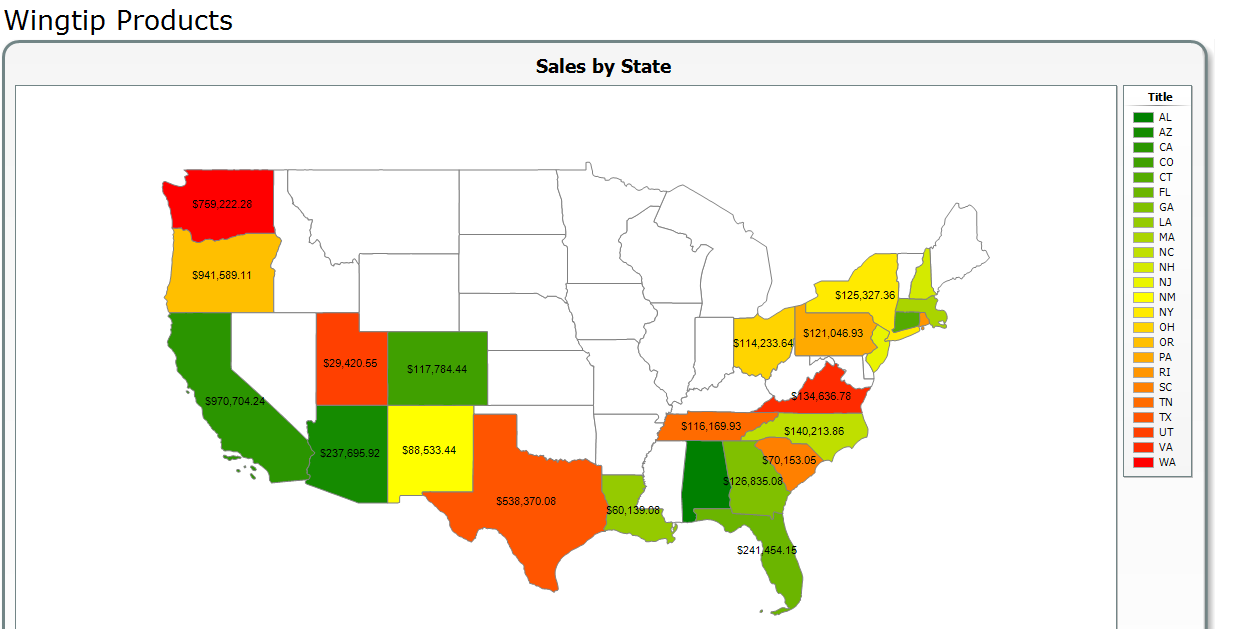
* 1. Modify **Set expression for: Label** to use **FormatCurrency** as displayed in the figure below then click **OK**.



1. Your map will now display **<<Expr>>** as the value of the states in design mode.



1. Run the report to see if the values are now displaying in currency format.



1. Save the report and retest the report in the browser to ensure all is okay.

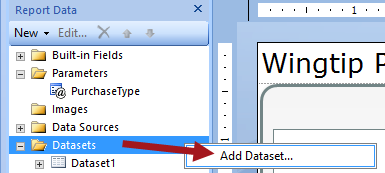
### Exercise 3: Add Parameters to Map Report

In this exercise you will set parameters on the SalesbyStateReport to filter by Purchase Type.

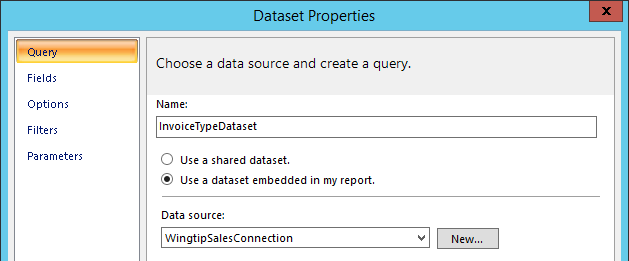
1. Go back to the **SalesbyStateReport** in Report Builder.

#### Create Dataset for Invoice Type

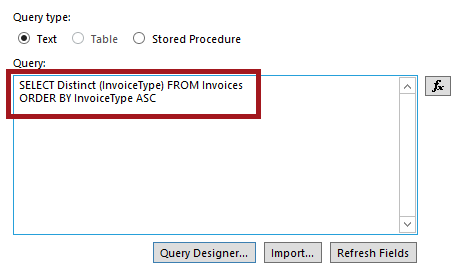
1. Make a new Dataset to Query the InvoiceType:
   1. In the **Report Data** pane, right-click on **Datasets** and click **Add Dataset…**



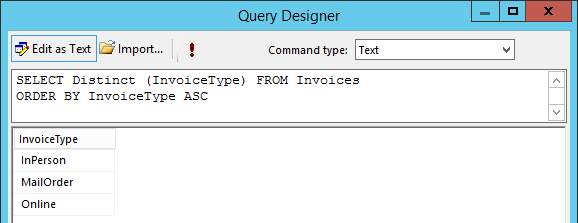
* 1. In the **Name** textbox type **InvoiceTypeDataset**.
  2. Select the option **Use a dataset embedded in my report** and then in the drop-down list select **WingtpSalesConnection**.



* 1. In the Query text box type: **SELECT Distinct (InvoiceType) FROM Invoices ORDER BY InvoiceType ASC**

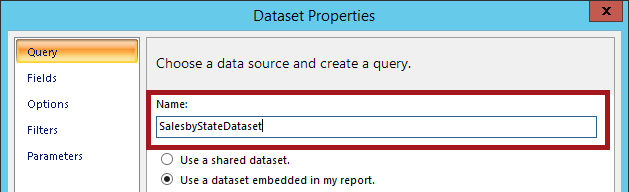


* 1. To test, click **Query Designer…** and then click the red exclamation **!** to run. You should see the same values as shown in the figure below.

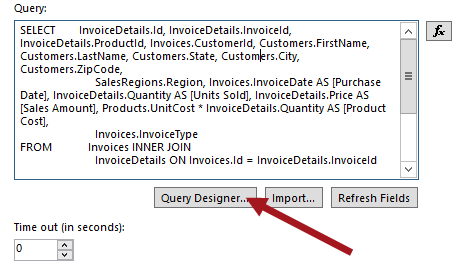


#### Modify Dataset1

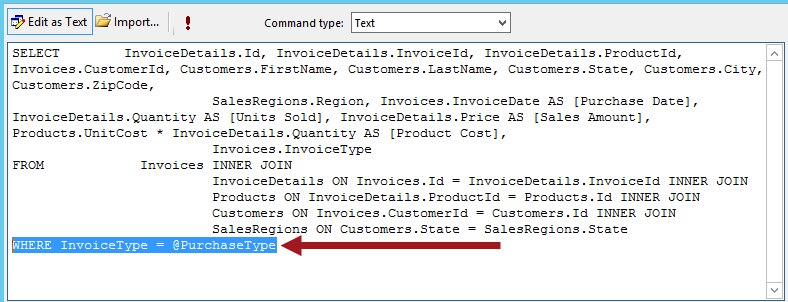
1. Make the following updates from the Query properties panel:
   1. In the **Report Data** pane, expand **Datasets** and then right-click on **DataSet1** and select **Dataset** **Properties**.
   2. Change the **Name** from **Dataset1** to **SalesbyStateDataset**.



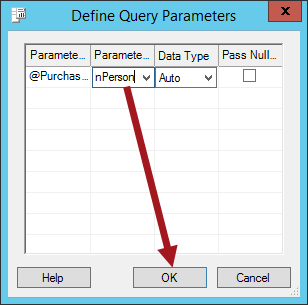
* 1. Click on **Query Designer**.



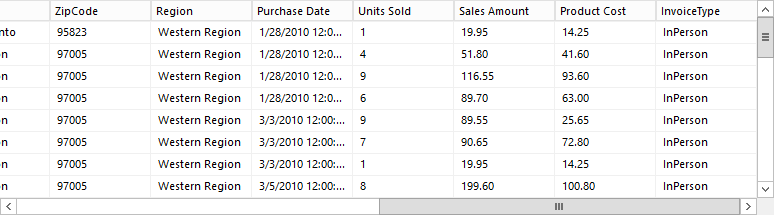
* 1. Scroll down to the bottom of the query and insert a new line **WHERE InvoiceType = @PurchaseType**.



* 1. Click the red exclamation **!** to test.
  2. Type **InPerson** in the Parameter value and then click **OK**.



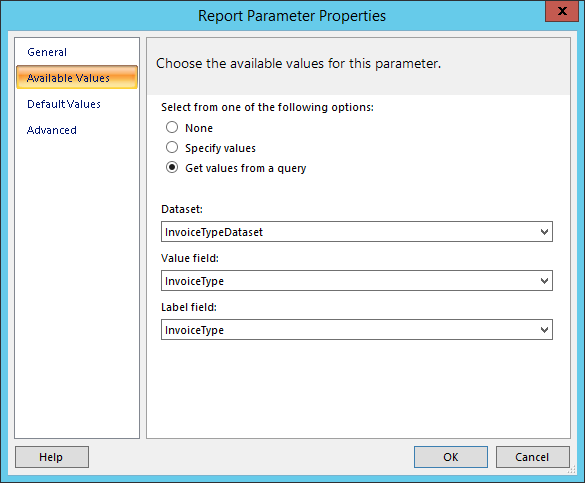
* 1. The data returns showing only items that have a value of **InPerson** for the **InvoiceType**.



* 1. Click **OK** and then click **OK** again.

**PurchaseType** is automatically added to the **Parameters**

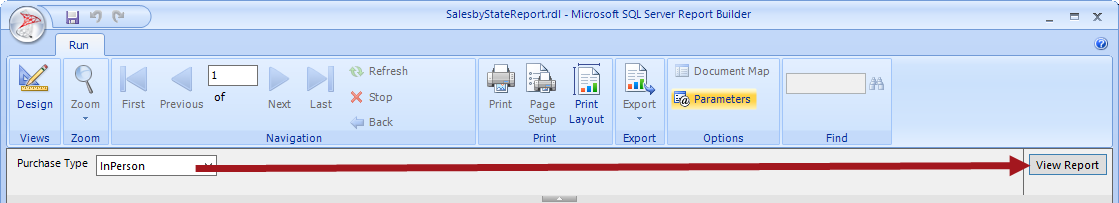
1. Modify the **PurchaseType** parameter by following the steps below:
   1. In the **Report Data** pane, expand **Parameters** and then double-click on **PurchaseType**.
   2. Now click on **Available Values** in the left pane.
   3. Select the **Get values from a query** option. Then choose **InvoiceTypeDataset** from the Dataset drop-down list. **InvoiceType** for **Value** field and **InvoiceType** for **Label** field.



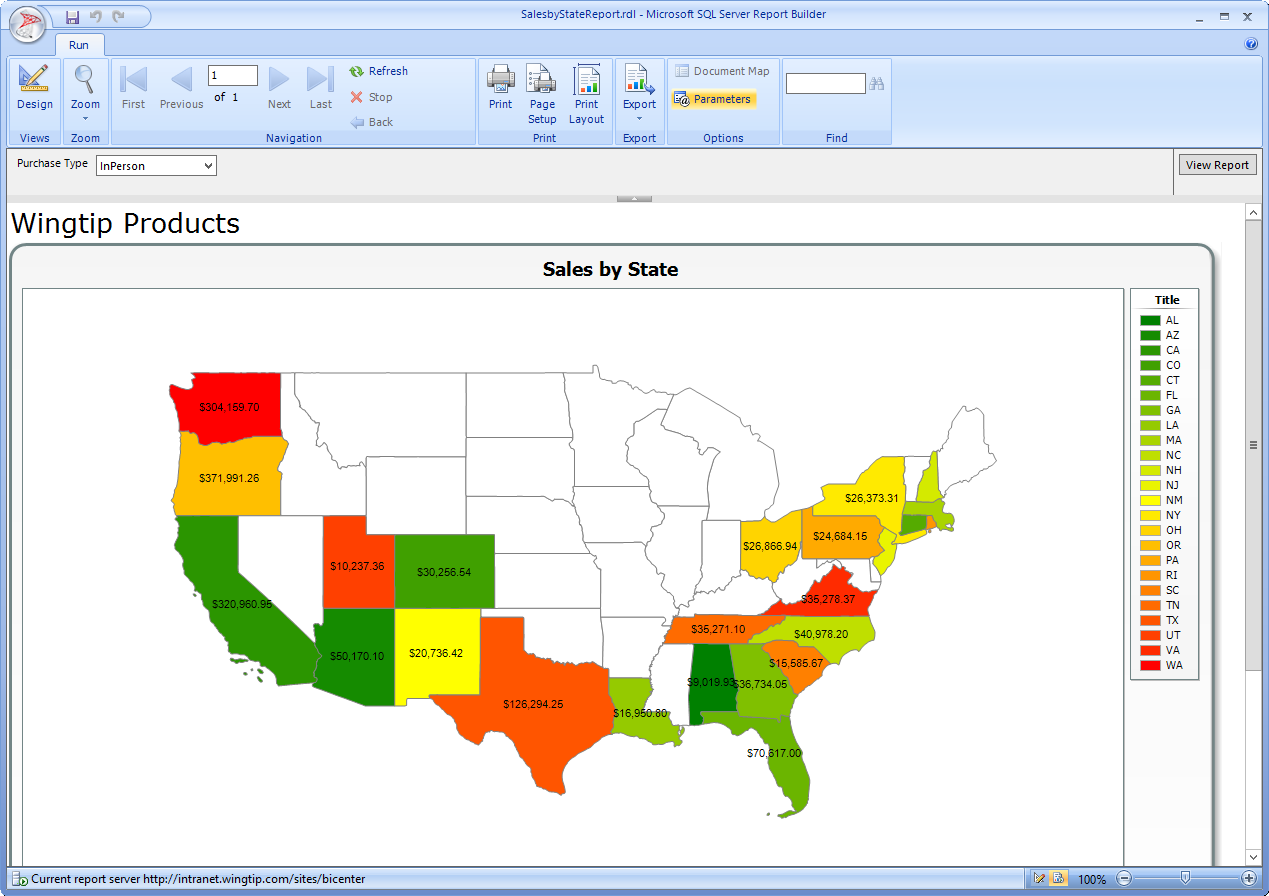
1. Click **OK**.
2. **Save** the report.

Saving the report now is a good idea because sometimes Report Builder will crash when you run reports for testing.

1. Test the report.
   1. Click the **Run** ribbon button to test the report.
   2. Select a value for the **Purchase Type** and then click **View Report**.



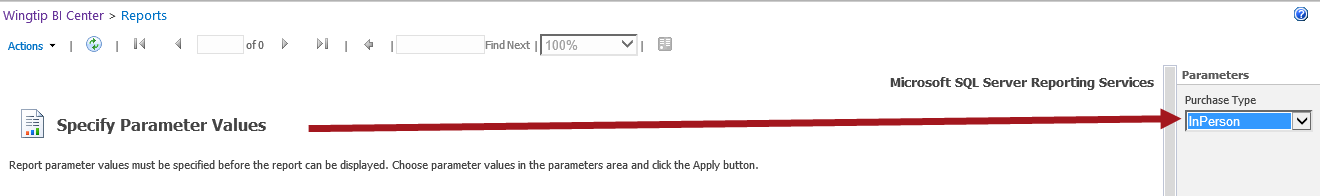
* 1. The report will now render the data based on the Purchase Type you selected.



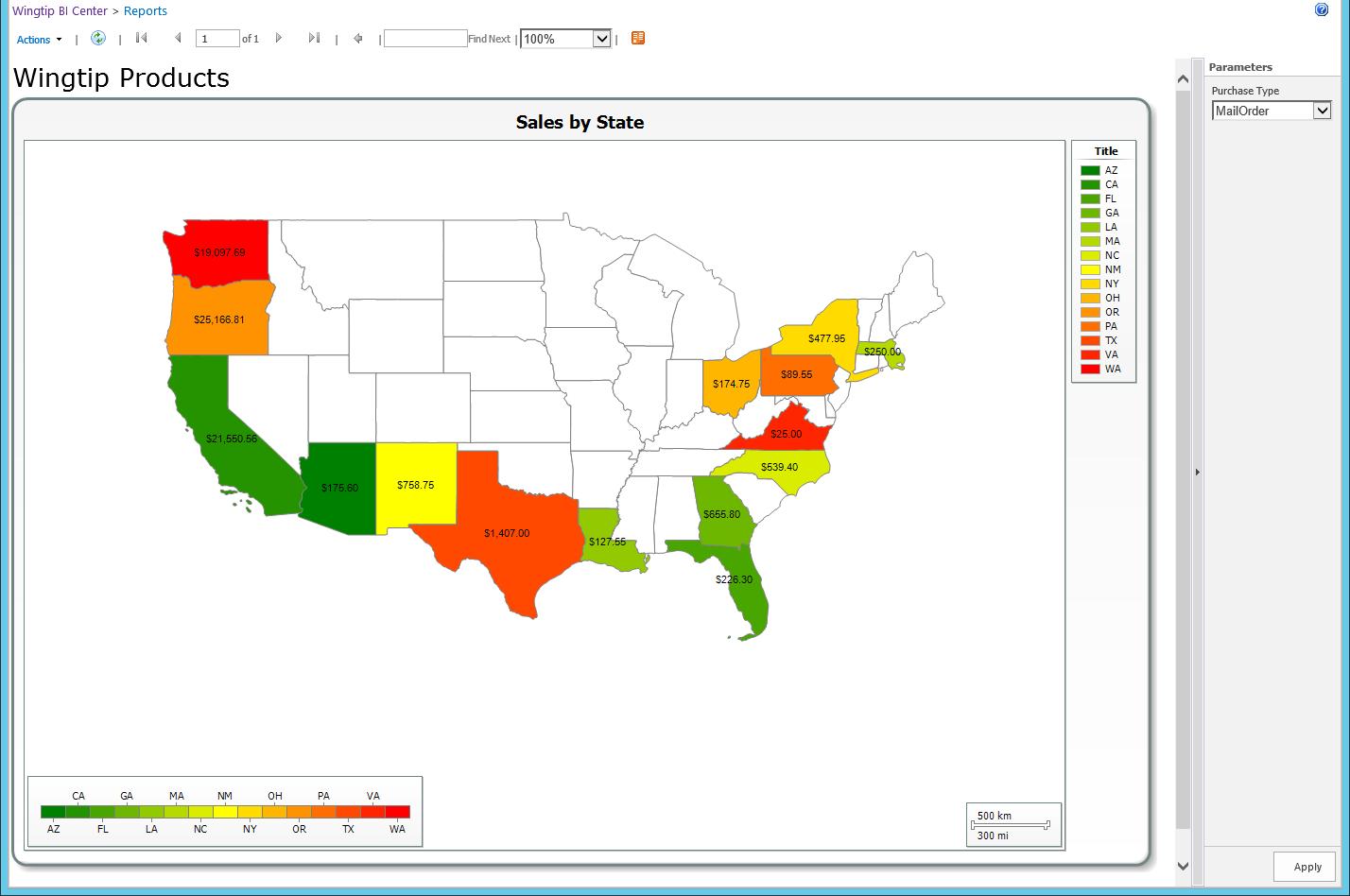
* 1. **Save** the report and then close **Report Builder**.

#### Test Report in Browser

1. Go back to the **SalesReport** in the BI Center site and refresh the report.
2. Set the **Purchase Type** Parameter and then click **Apply**.



1. The report displays with the filter Parameters applied. Change the Purchase Type to a different value and notice the map changes.



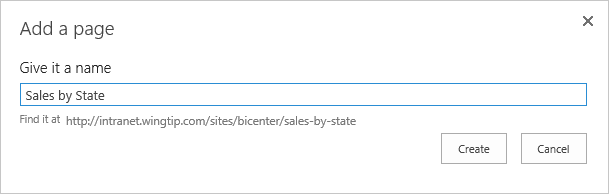
### Exercise 4: Using the Report Viewer Web Part

In this exercise you create a Web Part page, add a Report Viewer Web Part, add a Filter Web Part, and customize the Report Viewer Web Part.

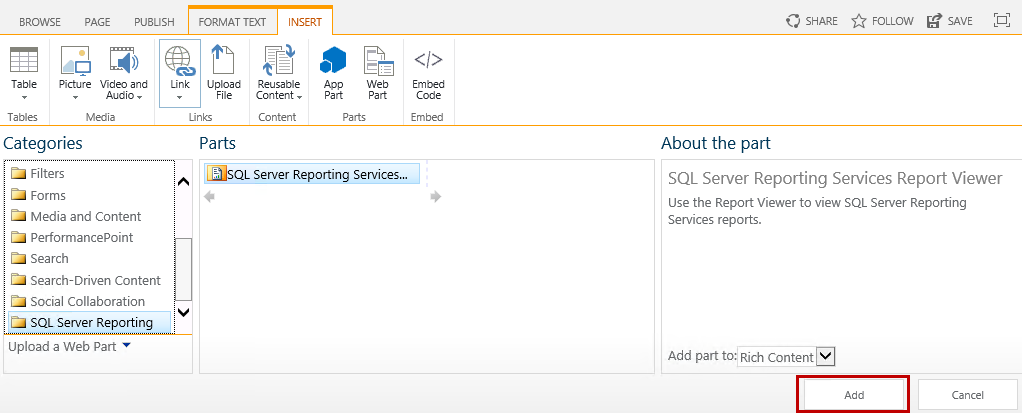
1. Open the Business Intelligence Center site root: [**http://intranet.wingtip.com/sites/bicenter**](http://intranet.wingtip.com/sites/bicenter)

#### Add Report View Web Part to Web Part Page

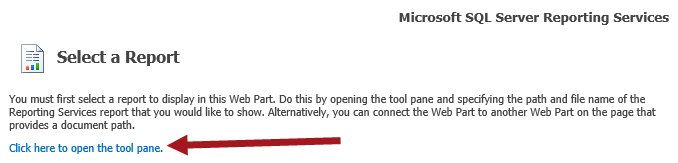
1. Click on the **Site Actions icon** and select **Add a page**.
2. Type **Sales by State** for the Name and click **Create**.



1. The page is created and you should be in edit mode. From the **Insert** tab, click **Web** **Part**.
2. Select **SQL Server Reporting** from the **Categories**, select **SQL Server Reporting Services Report Viewer** and then click **Add**.



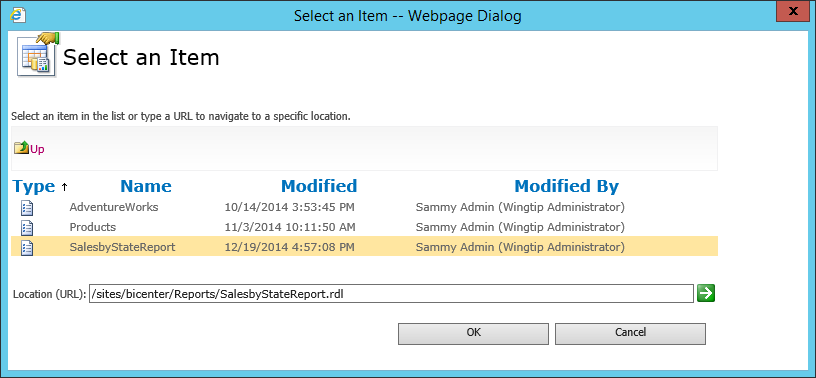
1. Click on the **Click here to open the tool pane** link.

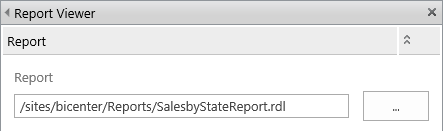


1. Select the **Ellipse(…)** button for **Report**.

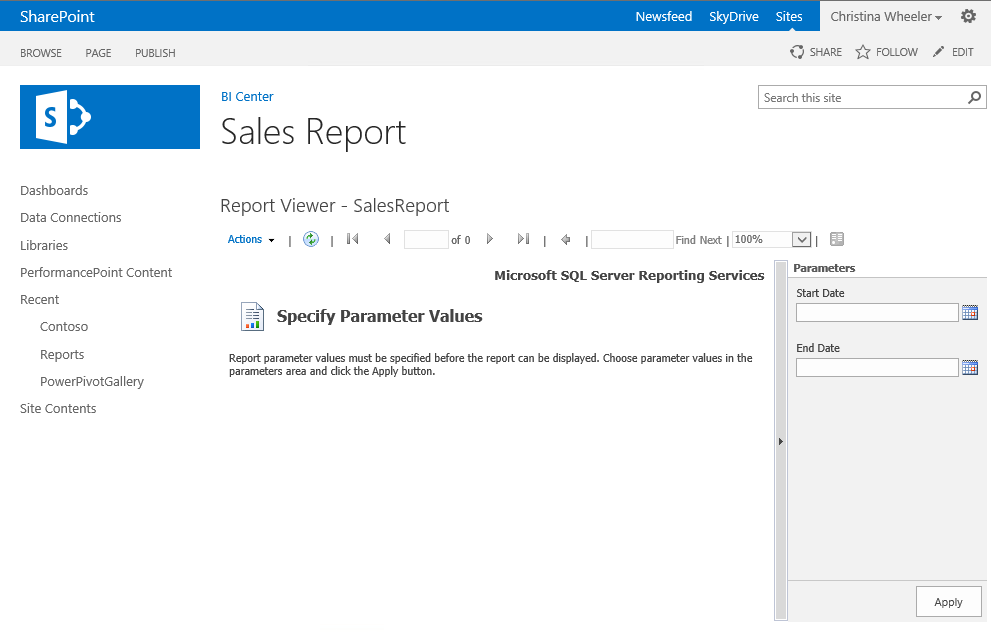


1. Navigate back to the **Reports** library and then select the **SalesbyStateReport** and click **OK**.



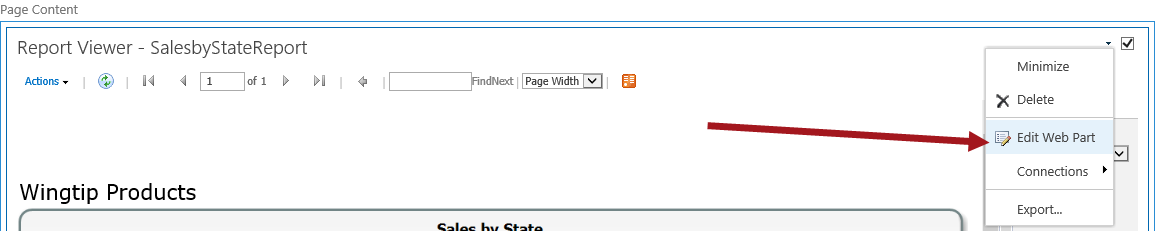


1. Scroll to the bottom of the Web Part edit panel and then click **OK**.
2. Publish the page.
   1. From the **Publish** tab, click the **Publish** ribbon button.

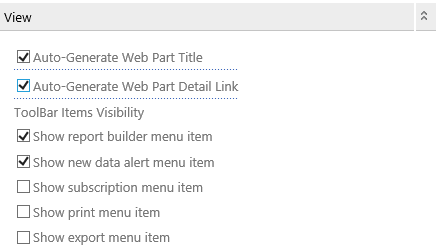


#### Customize the Report Viewer Web Part

1. To get into edit mode for the page, from the **Page** tab:
   1. Click the **Check Out** ribbon button.
   2. Click the **Edit** ribbon button.
2. Hover over the upper-right corner of the **Report Viewer** web part and select **Edit Web Part**.



1. Expand the **View** section to set the properties for the display. Uncheck properties you don’t not want to display and make any other desired changes. Click **Apply** to view your changes and click **OK** once complete.



1. Publish the page.
   1. From the **Publish** tab, click the **Publish** ribbon button.

You can continue to make changes to the report modifying the size, formatting, etc. until you get the desired layout you want. Once you have published the page, you have completed all the exercise in this lab.