Configuring Datasource Refresh and DirectQuery

Lab Time: 60 minutes

Lab Folder: C:\Student\Modules\08_Gateways\Lab

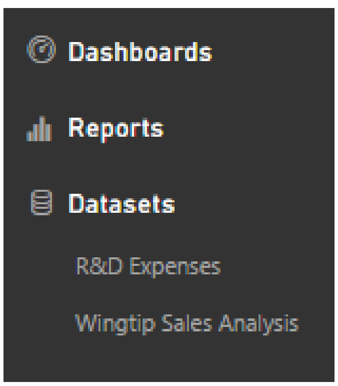
Lab Overview: In this lab you will get hands-on experience configuring datasets in the Power BI service for the manual and scheduled refreshing of data from underlying data sources. You will also get experience installing and configuring a Power BI gateway to refresh data from an on-premises data source. In the final experience, you will create a Power BI Desktop project based on a live connection. This will give you a chance to see how different things are when you are creating queries and data modeling in DirectQuery mode.

Lab Dependency: This lab assumes you completed lab 5 titled **Designing Interactive Reports in Power BI Desktop** in which you created and published a Power BI Desktop project named **Wingtip Sales Analysis.pbix**. If you would like to begin work on this lab without completing lab 6, copy the lab solution file named **Wingtip Sales Analysis.pbix** which is located in the student folder at **C:\Student\Modules\06_Reports\Lab\Solution** into the folder at **C:\Student\Projects**.

Exercise 1: Configure Data Source Credentials in the Power BI Service

In this exercise you will create a new dashboard using the dataset and report you created in early labs using Power BI Desktop.

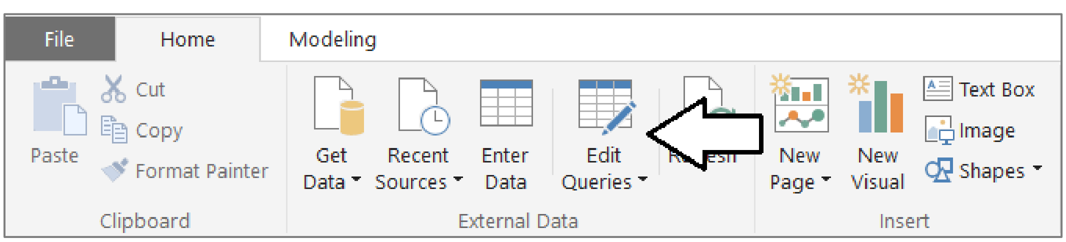
1. Using the browser log into the Power BI service and access your personal workspace. While your personal workspace might contain many different dashboards, reports and datasets by this point, you should see two datasets named **R&D Expenses** and **Wingtip Sales Analysis** as shown in the following screenshot.



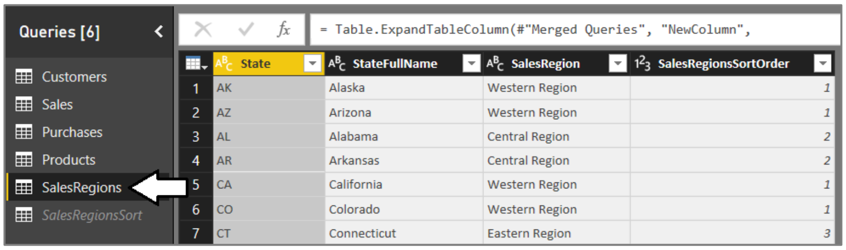
2. Launch Power BI Desktop and open the PBIX project file in the **C:\Student\Projects** folder named **Wingtip Sales Analysis.pbix**.

Currently, the Power BI Desktop project named **Wingtip Sales Analysis.pbix** contains a data source based on a local Excel workbook file located at **C:\Student\Data\WingtipSalesRegions.xlsx**. In the next step you will remove this local file dependency by updating the data source to use copy of the same Excel workbook that can be access from across the Internet using anonymous access.

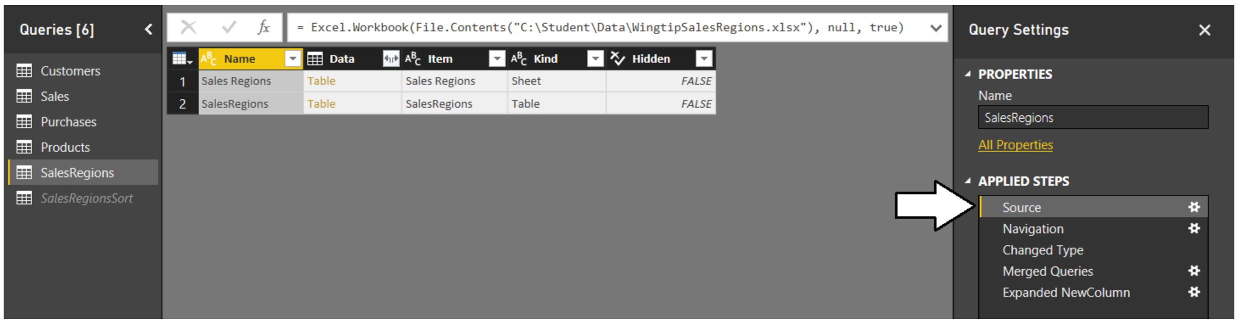
3. Update the data source for the query named **SalesRegions**.
   a) Once the project has opened, click the **Edit Queries** button to display the Query Editor window.



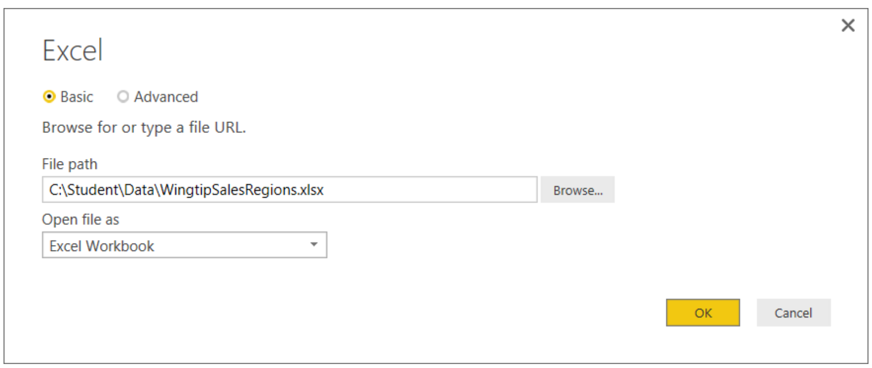
   b) In the **Queries** list on the left, select the query named **SalesRegion**.



c) Inside the **APPLIED STEPS** list on the right, locate the **Source** step and click its gear icon button to modify its value.



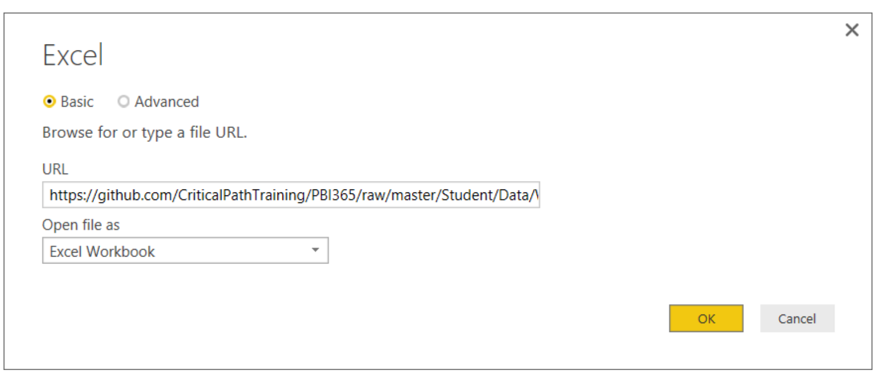
d) You should be prompted with the **Excel** dialog which displays the current file path.



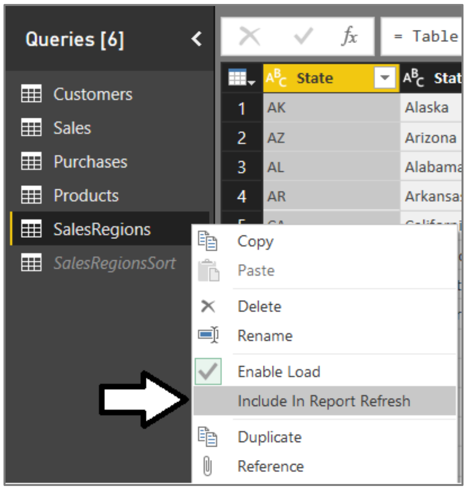
e) Update the **File path** value for **WingtipSalesRegions.xlsx** by copying and pasting the following URL.

https://github.com/CriticalPathTraining/PBI365/raw/master/Student/Data/WingtipSalesRegions.xlsx

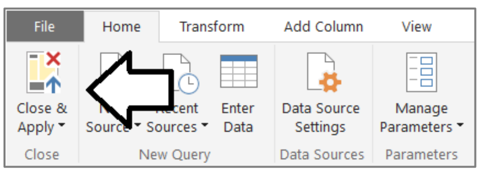
f) Once the **Excel** dialog matches the screenshot below, click **OK** to save your changes.



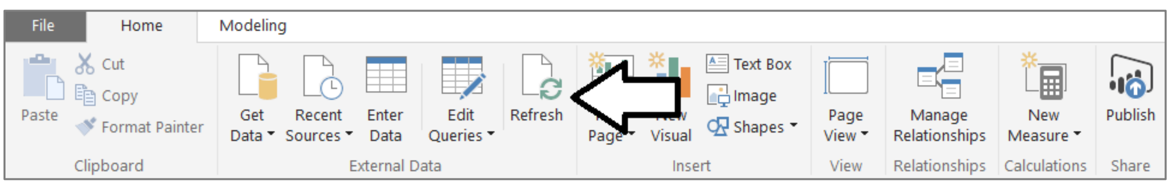
g) In the **Queries** list on the left, right-click on the query named **SalesRegions** and then click the **Include In Report Refresh** option once to put this property into a disabled state. Once you have disabled **Include in Report Refresh** property, you should be able to verify that this property has no checkmark next to it as shown in the following screenshot.



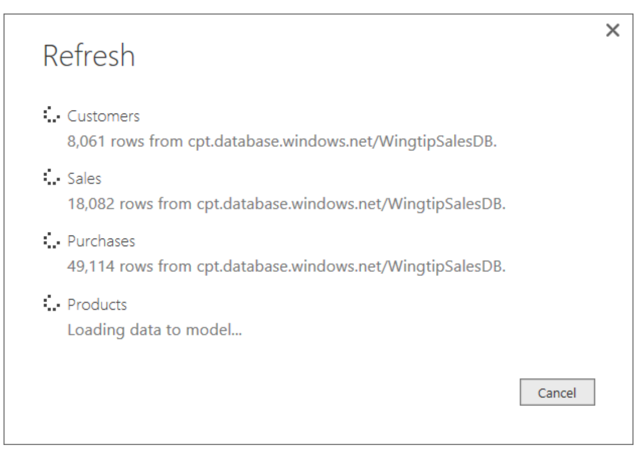
h) Click the **Close & Apply** button to close the Query Editor window.



i) Click the **Refresh** button on the **Home** ribbon tab of the main Power BI desktop application window to refresh the project's data sources.



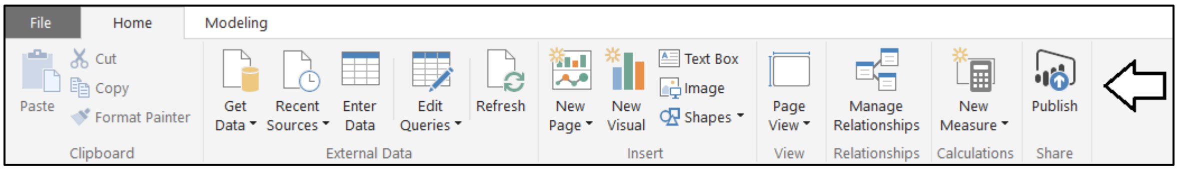
j) You should see all the tables from the Azure SQL databases being refreshed. However, the SalesRegions query was not refreshed because you set its **Include in Report Refresh** property to a disabled state.



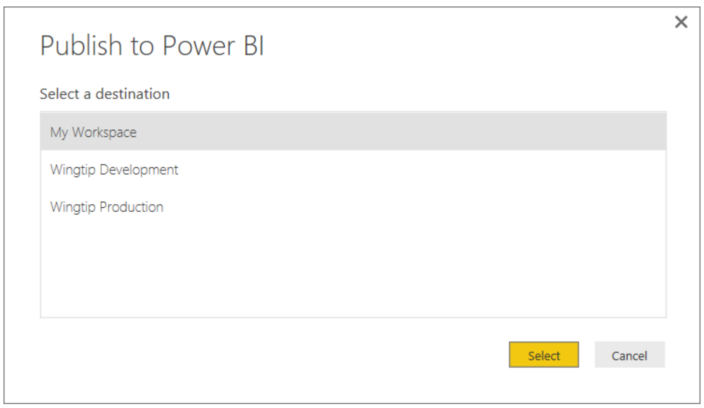
k) Click the **Save** button to save your changes to the project named **Wingtip Sales Analysis.pbix**.

4. Publish your project to push the changes you made to **Wingtip Sales Analysis.pbix** into the Power BI service.

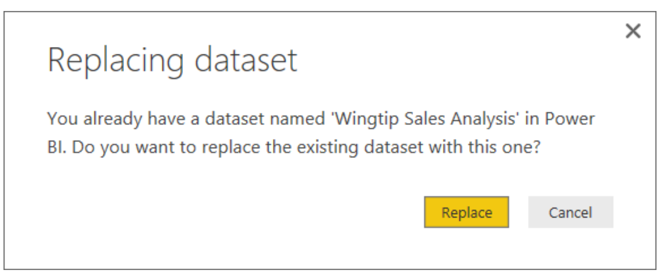
a) Click the **Publish** button in the **Home** tab of the ribbon.



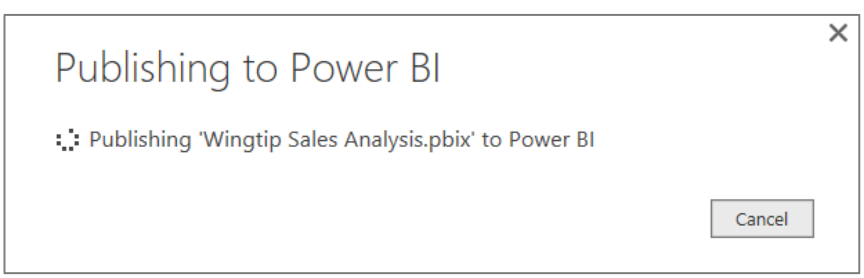
b) When prompted to **Select a destination**, choose **My Workspace** then click the **Select** button.



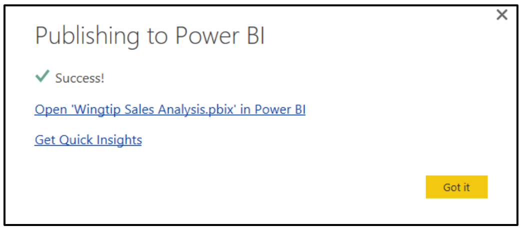
c) If you are prompted with the **Replacing dataset** dialog, click the **Replace** button to continue.



d) Wait while Power BI Desktop publishes your project updates to the Power BI service.

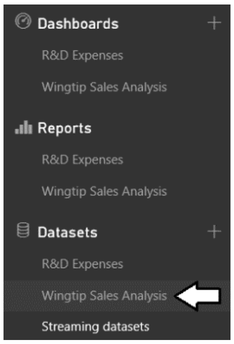


e) When Power BI Desktop is done, it should display a message indicating success and a link to navigate to your personal workspace in the browser. Click the link with the caption **Open 'Wingtip Sales Analysis' in Power BI**.

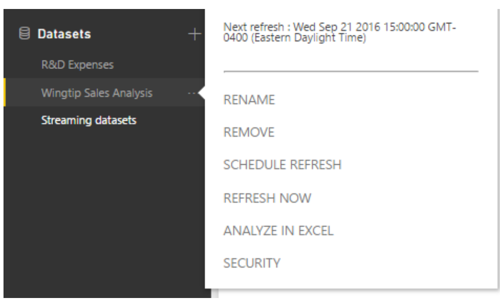


5. Try and perform a manual refresh on the **Wingtip Sales Analysis** dataset.

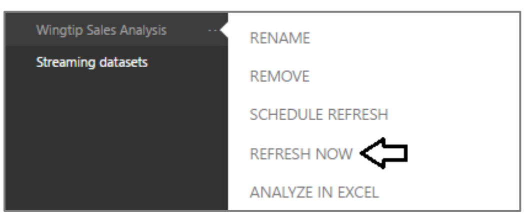
a) At this point, you should see the dashboards, reports and datasets in your personal workspace including the datasets named **Wingtip Sales Analysis**.



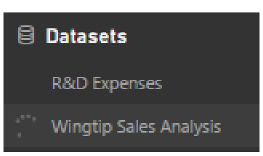
b) Click the ellipse button (…) to the right of the **Wingtip Sales Analysis** dataset to see the available dataset commands.



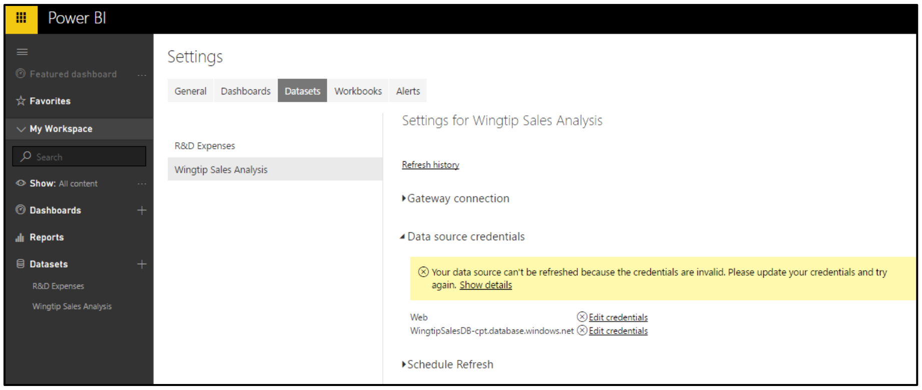
c) Click the **REFRESH NOW** command on **Wingtip Sales Analysis** dataset.



d) You will see a spinning animation to the left of the dataset indicating that the Power BI service is attempting to refresh its data.



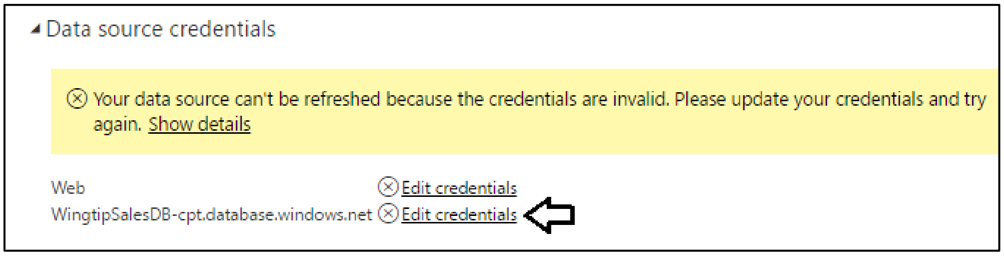
e) The manual update should fail because you have not yet updated the credentials for the dataset's underling data sources. When you attempt a refresh and it fails, the Power BI service should automatically redirect you to the dataset properties page for the **Wingtip Sales Analysis** dataset and display an error message in the **Data source credentials** section.



Note that this lab exercise makes the assumption that you have not already configured the credentials for the underlying data sources. If you have already configured credentials for the SQL Azure database, you might find that the manual refresh attempt succeeds instead of failing as indicated in the previous step.

6. Configure the credentials for the data source associated with the SQL Azure database.

a) In the **Data source credentials** section for the **Wingtip Sales Analysis** dataset, click the **Edit credentials** link for the data source named **WingtipSalesDB-cpt.database.windows.net**.



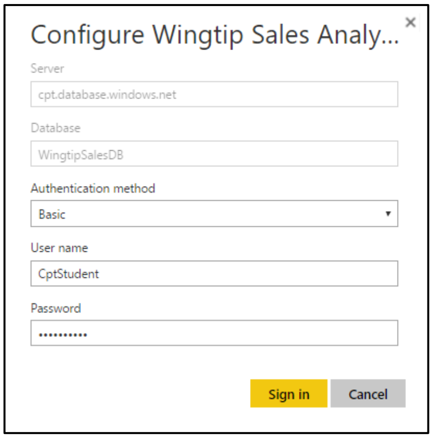
b) You are prompted with a dialog to configure the credentials for this data source.

c) Change the **Authentication method** to **Basic**.

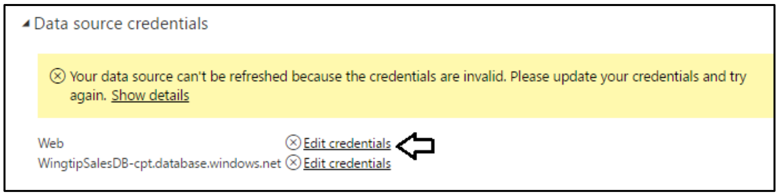
d) Enter a **User name** of **CptStudent**.

e) Enter a **Password** of **pass@word1**.

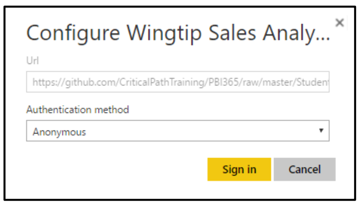
f) Click the **Sign In** button to save the credentials and cache them inside the Power BI service.



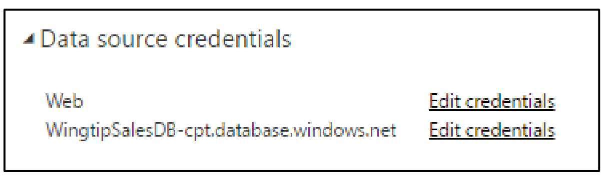
7. Configure the data source credentials for the Excel workbook with sales regions located in a public GitHub repository.
   a) Click the **Edit credentials** link for the data source named **Web**.



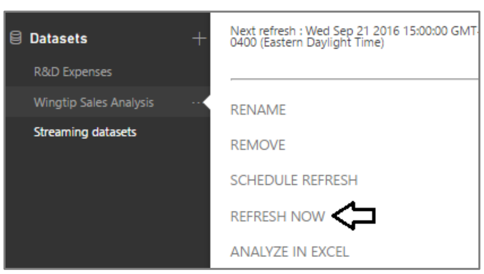
   b) You are prompted with a dialog to configure the credentials for this data source.
   c) Change the **Authentication method** to **Anonymous**.
   d) Click the **Sign In** button to save the credentials (*or lack thereof*) for this data source.



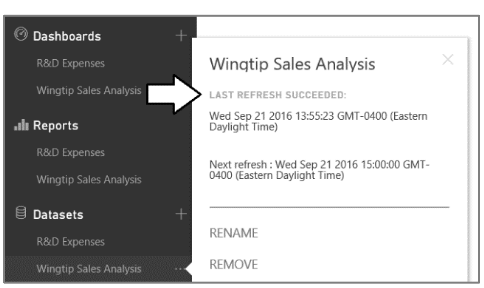
   e) At this point, the error message should no longer be displayed in the **Data source credentials** section.



8. Attempt to perform a manual refresh a second time now that you have updated the dataset's underlying data source credentials.
   a) Click the **REFRESH NOW** command on **Wingtip Sales Analysis** dataset.



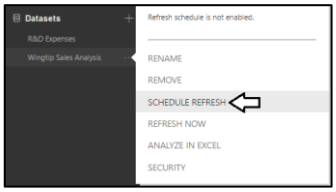
   b) You will see a spinning animation to the left of the dataset which indicates the Power BI service is attempting to refresh the underlying data. When the spinning animation stops, expand the dataset's flyout menu and verify that the refresh operation was successful.



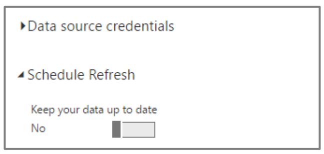
Exercise 2: Create a Schedule to Automatically Refresh a Dataset

In the previous exercise you configured the data source credentials for the **Wingtip Sales Analysis** dataset and then you executed the command to perform a manual refresh. In this exercise you will configure the **Wingtip Sales Analysis** dataset to automatically refresh on a periodic schedule

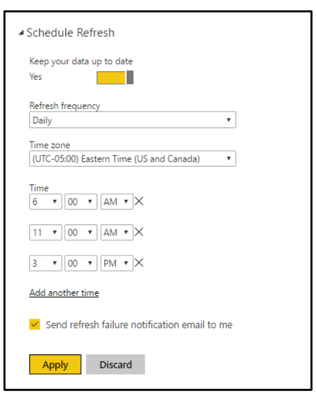
1. Click the ellipse flyout menu for the **Wingtip Sales Analysis** dataset and click the **SCHEDULE REFRESH** command. When you do this, you should be redirected to a page that displays the **Schedule Refresh** section for that dataset.



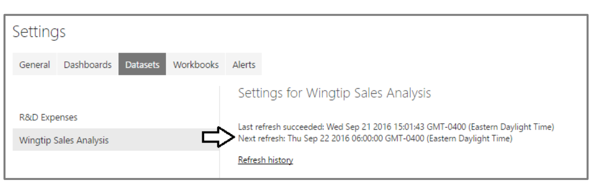
2. When you examine the **Schedule Refresh** section, you should be able to see that the **Keep your data up to date** property is set to **No** as its default setting.



3. Configure the **Schedule Refresh** settings for the **Wingtip Sales Analysis** dataset.
   a) Update the **Keep your data up to date** property to **Yes**.
   b) Update the **Refresh frequency** property to **Daily**.
   c) Set the time to 6:00 AM.
   d) Click the **Add another time link** and configure the second time for **11:00 AM**.
   e) Click the **Add another time link** and configure the third time for **3:00 PM**.
   f) Make sure the **Send refresh failure notification email to me** checkbox is in a checked state.
   g) When your **Schedule Refresh** settings match the following screenshot, click the **Apply** button to save your changes.



   h) Once you have configured the scheduled refresh settings, you should be able to verify when the next refresh is scheduled.



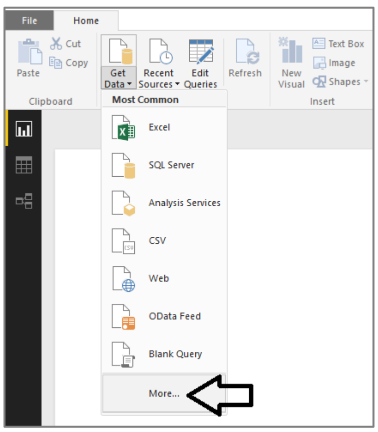
Exercise 3: Connect to a Data Source using DirectQuery Mode

In this exercise you will create a new Power BI Desktop project that uses DirectQuery mode to retrieve data from the SQL Azure database named **WingtipSalesDB**. Note that this is the same SQL Azure database that you have worked with in earlier lab exercises. As you will see, working in DirectQuery mode usually requires writing SQL statements for queries to compensate for the data modeling limitation which prevents you from creating calculated columns.

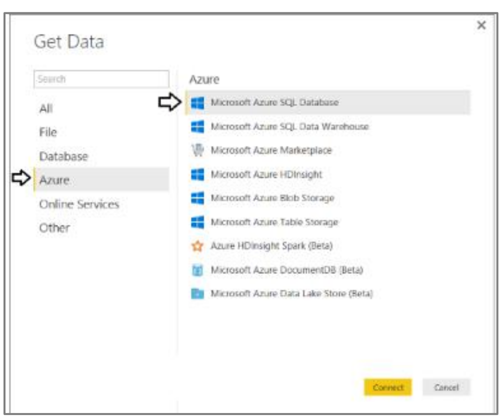
1. Launch Power BI Desktop to start a new project.
2. Save the new project as **DirectQuery Lab.pbix** using the following path.

```
C:\Student\Projects\DirectQuery Lab.pbix
```

3. Create a new query in DirectQuery mode to retrieve customer data from an Azure SQL database.
   a) Drop down the **Get Data** menu button on the ribbon and click **More…**.



   b) In the **Get Data** dialog…
      i) Select **Azure** from the list on the left.
      ii) Select **Microsoft Azure SQL Database** from the list on the right.
      iii) Click the **Connect** button.



   c) When you are prompted with the **SQL Server Database** dialog, complete the following tasks.
      i) Enter a **Server** value of **cpt.database.windows.net**
      ii) Enter a **Database** value of **WingtipSalesDB**
      iii) Select the option button for **DirectQuery**.
      iv) Expand the **Advanced options** section.

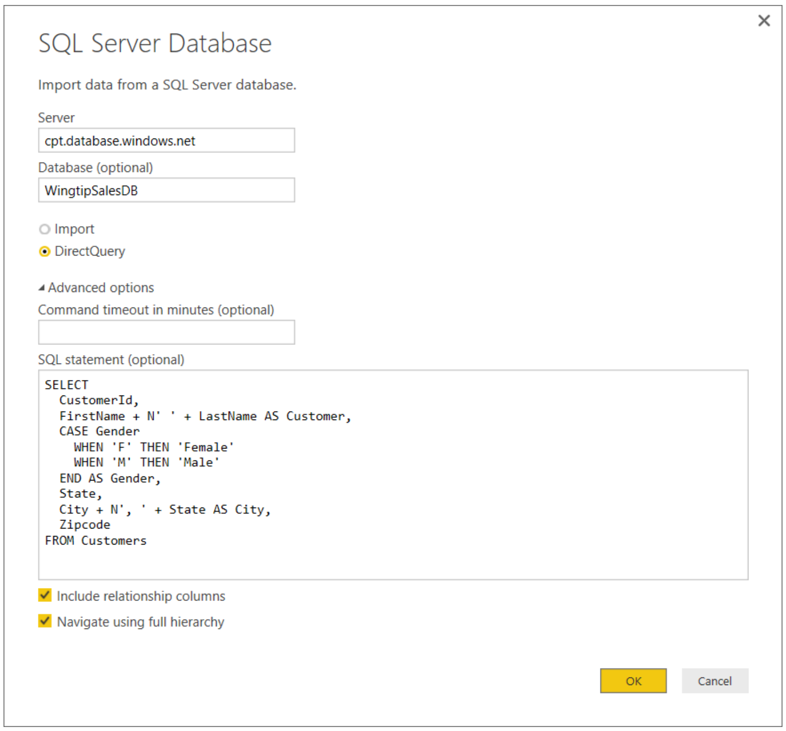
d) In the **SQL statement** textbox inside the **Advanced option** section, copy and paste the following SQL statement.

```
SELECT
  CustomerId,
  FirstName + N' ' + LastName AS Customer,
  CASE Gender
    WHEN 'F' THEN 'Female'
    WHEN 'M' THEN 'Male'
  END AS Gender,
  State,
  City + N', ' + State AS City,
  Zipcode
FROM Customers
```

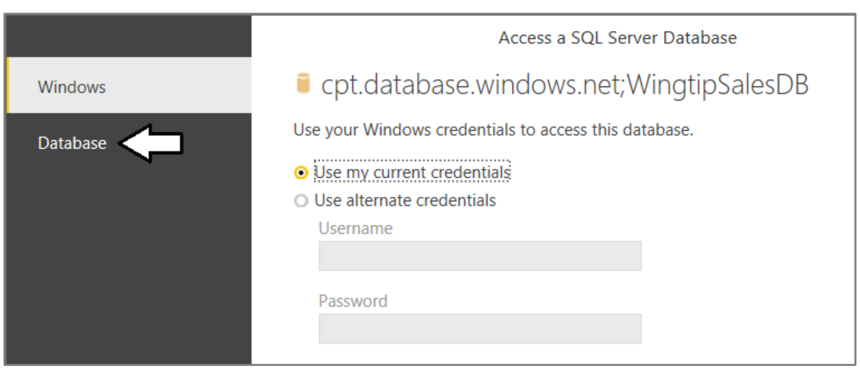
You can copy and paste this SQL statement from the text-based SQL file named **Customers.sql** located in the folder for this lab.

e) Make sure the checkboxes for **Include Relationship Columns** and **Navigate using full hierarchy** are both checked.

f) When the **SQL Server Database** dialog matches the following screenshot, click **OK**.



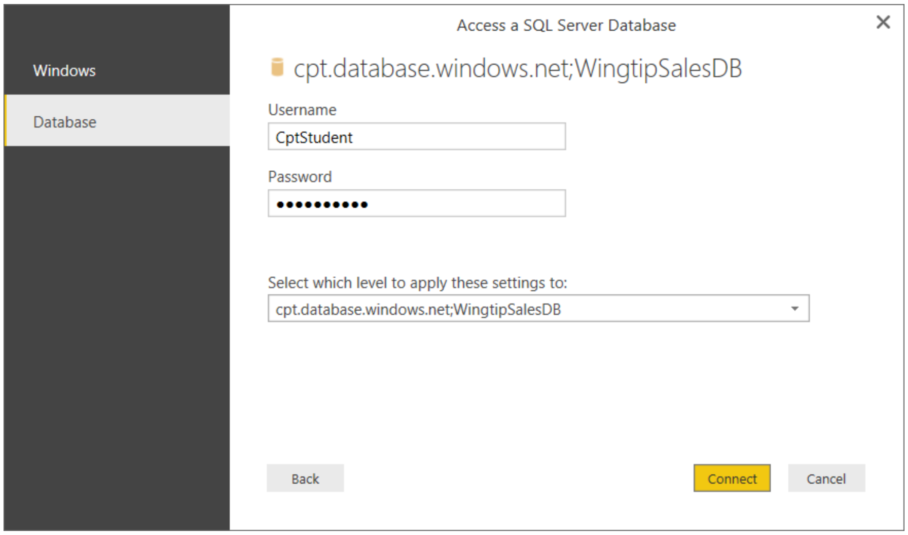
g) You should now be prompted with the **Access a SQL Server Database** dialog asking for data source credentials. Select the option for **Database** on the left side of the dialog.



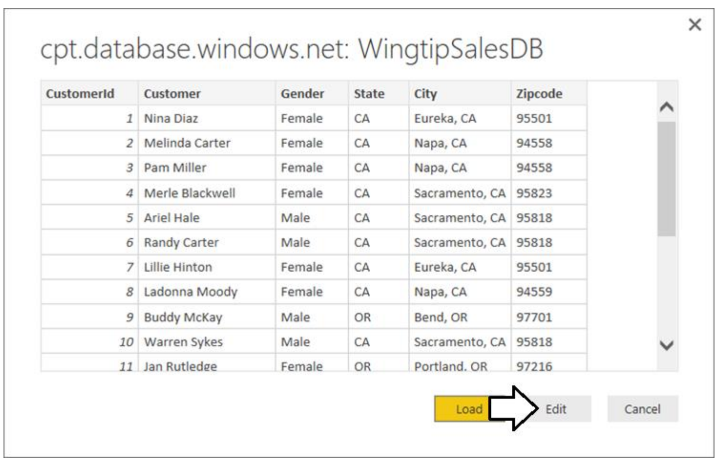
h) Enter a Username of **CptStudent**.

i) Enter a password of **pass@word1**.

j) Click the **Connect** button to submit the credentials for the Azure SQL database.

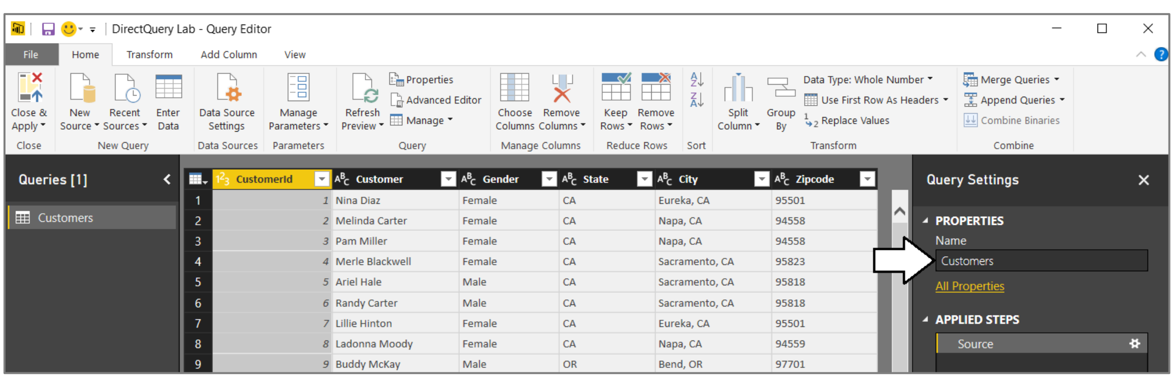


k) When you see the dialog which displays rows of data, click the **Edit** button to open the new query in the Query Editor window.



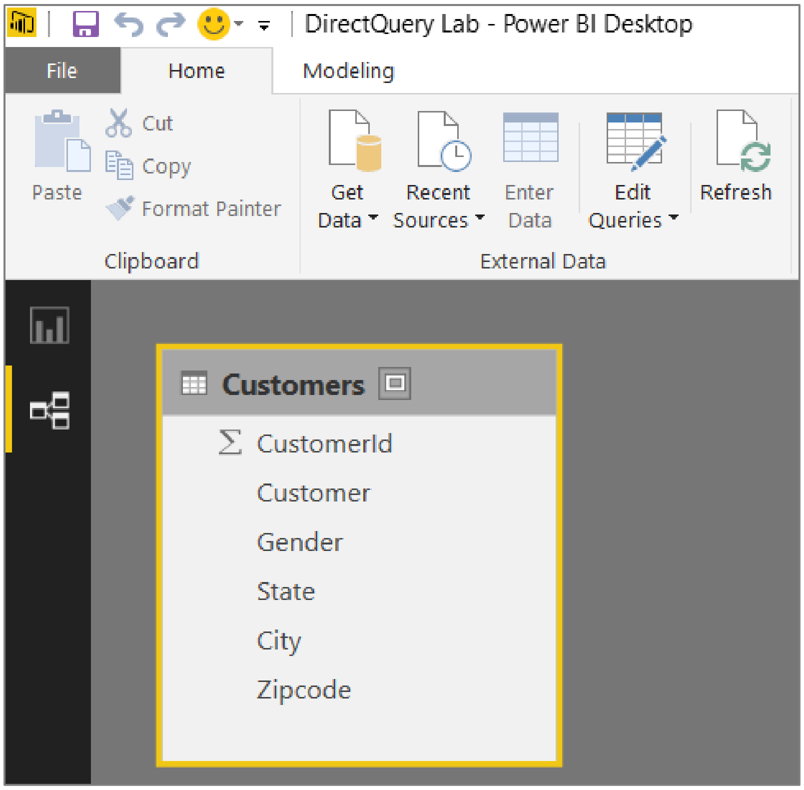
l) You should see that the new query has been created with the name **Query1**.

m) In the **Query Settings** pane in the right-side of the Query Editor window, change the query name from **Query1** to **Customers**.



n) Click the **Close & Apply** button to close the Query Editor window.

o) Navigate to Relationship View and inspect the **Customers** table.



You will notice that the left navigation has buttons to navigate to Report View and Relationship View. However, there is no navigation button for Data View because it is not supported when working in DirectQuery mode.

4. Save your work by clicking the **Save** button at the top left of the main Power BI Desktop window.
5. Create a new query in DirectQuery mode to retrieve product data from the same Azure SQL database.

a) Drop down the **Get Data** menu button on the ribbon and click **More…**.

b) On the **Get Data** dialog, select **Azure** in the list on the left. Next, select **Microsoft SQL Azure Database** on the right and then click the **Connect** button.

c) When you are prompted with the **SQL Server Database** dialog, complete the following tasks.

i) Enter a **Server** value of **cpt.database.windows.net**

ii) Enter a **Database** value of **WingtipSalesDB**

iii) Select the option button for **DirectQuery**.

iv) Expand the **Advanced options** section.

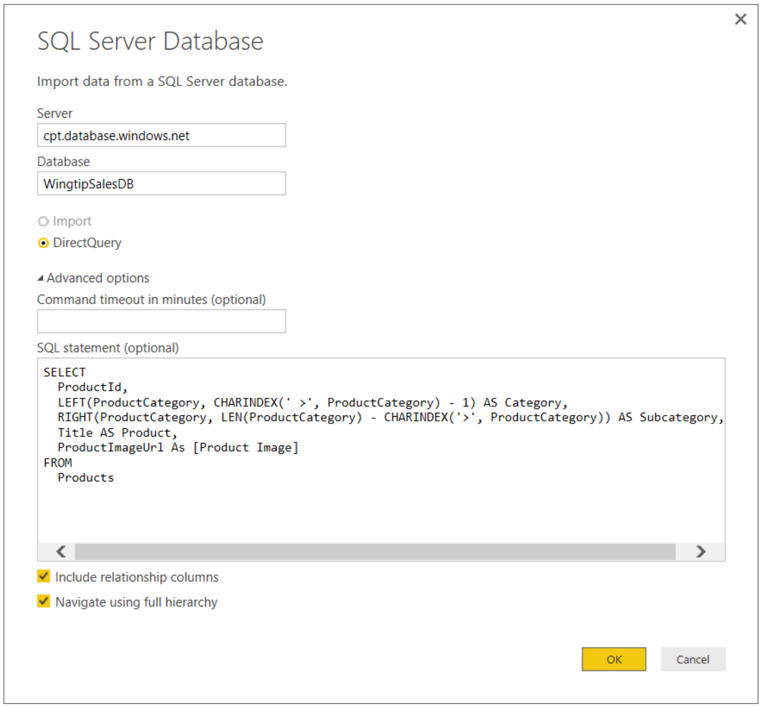
d) In the **SQL statement** textbox of the **Advanced options** section, copy and paste the following SQL statement.

```
SELECT
  ProductId,
  LEFT(ProductCategory, CHARINDEX(' >', ProductCategory) - 1) AS Category,
  RIGHT(ProductCategory, LEN(ProductCategory) - CHARINDEX('>', ProductCategory)) AS Subcategory,
  Title AS Product,
  ProductImageUrl As [Product Image]
FROM
  Products
```

You can copy and paste this SQL statement from the text-based SQL file named **Products.sql** located in the folder for this lab.

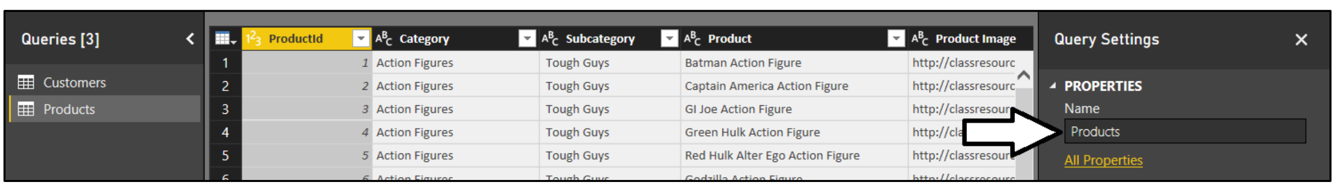
e) Make sure the checkboxes for **Include Relationship Columns** and **Navigate using full hierarchy** are both checked.

f) When the **SQL Server Database** dialog matches the following screenshot, click **OK**.



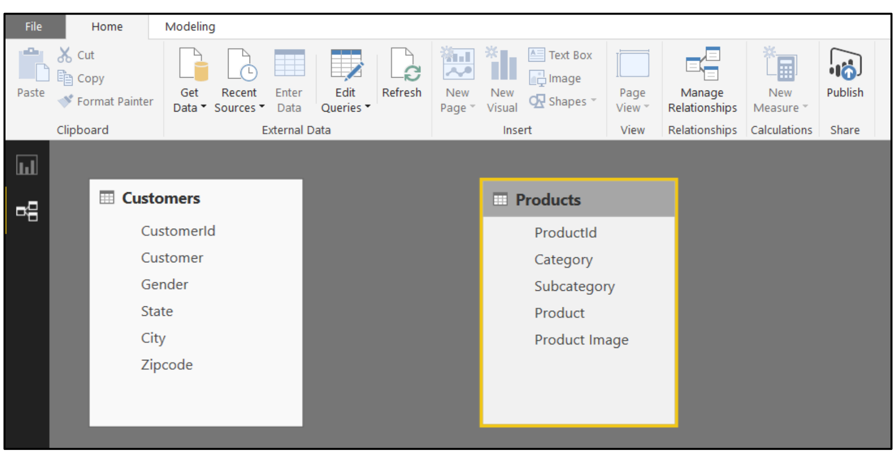
g) When you see the dialog which displays rows of data, click the **Edit** button to open the new query in the Query Editor window.

h) In the **Query settings** pane in the right-side of the Query Editor window, change the query name to **Products**.



i) Click the **Close & Apply** button to close the Query Editor window.

j) Navigate to Relationship View and inspect the **Products** table which should be displayed next to the **Customers** table.



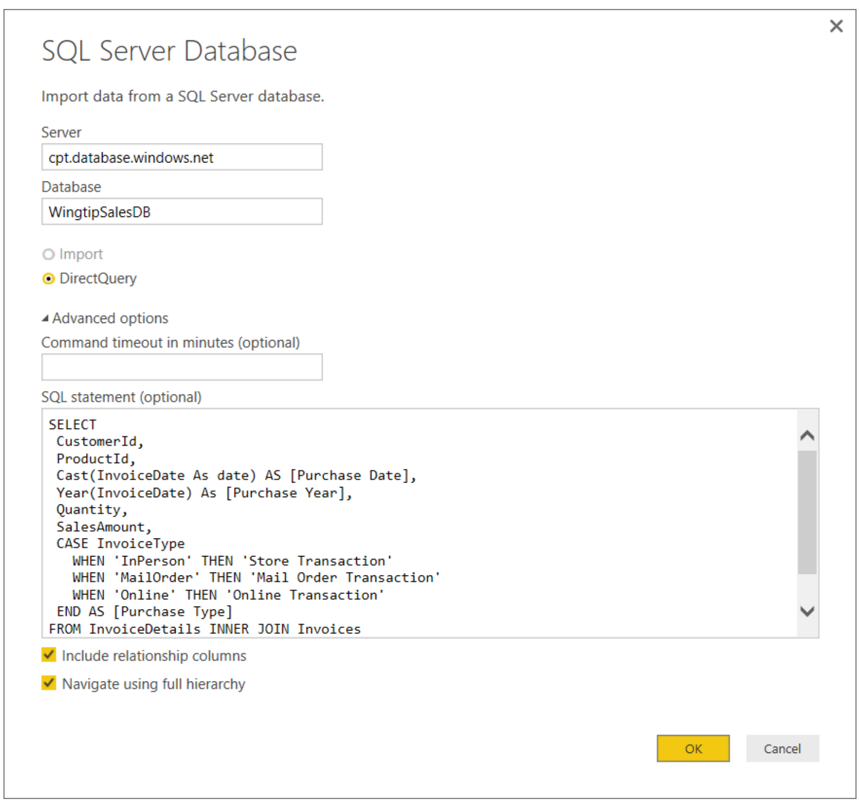
k) Save your work by clicking the **Save** button at the top left of the main Power BI Desktop window.

6. Create a new query in DirectQuery mode to execute a SQL statement which joins the **Invoice** table to the **InvoiceDetails** table.
   a) Drop down the **Get Data** menu button on the ribbon and click **More…**.
   b) On the **Get Data** dialog, select **Azure** in the list on the left. Next, select **Microsoft SQL Azure Database** on the right and then click the **Connect** button.
   c) When you are prompted with the **SQL Server Database** dialog, complete the following tasks.
      i) Enter a Server value of **cpt.database.windows.net**
      ii) Enter a Database value of **WingtipSalesDB**.
      iii) Select the option button for **DirectQuery**.
      iv) Expand the **Advanced options** section.
   d) In the **SQL statement** textbox of the **Advanced options** section, copy and paste the following SQL statement.

```
SELECT
 CustomerId,
 ProductId,
 Cast(InvoiceDate As date) AS [Purchase Date],
 Year(InvoiceDate) As [Purchase Year],
 Quantity,
 SalesAmount,
 CASE InvoiceType
   WHEN 'InPerson' THEN 'Store Transaction'
   WHEN 'MailOrder' THEN 'Mail Order Transaction'
   WHEN 'Online' THEN 'Online Transaction'
 END AS [Purchase Type]
FROM InvoiceDetails INNER JOIN Invoices
  ON InvoiceDetails.InvoiceId = Invoices.InvoiceId
```

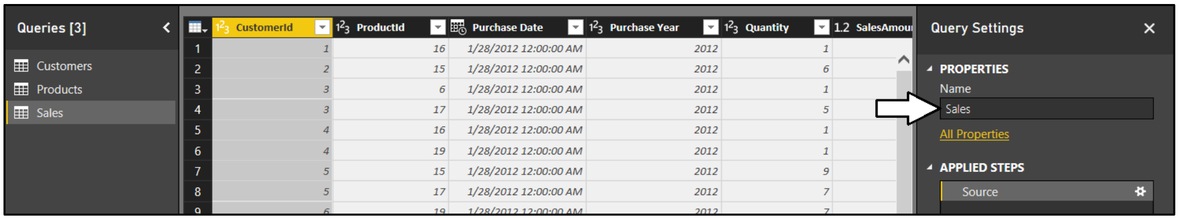
You can copy and paste this SQL statement from the text-based SQL file named **Sales.sql** located in the folder for this lab.

   e) Make sure the checkboxes for **Include Relationship Columns** and **Navigate using full hierarchy** are both checked.
   f) When the **SQL Server Database** dialog matches the following screenshot, click **OK**.



g) When you see the dialog which displays rows of data, click the **Edit** button to open the new query in the Query Editor window.

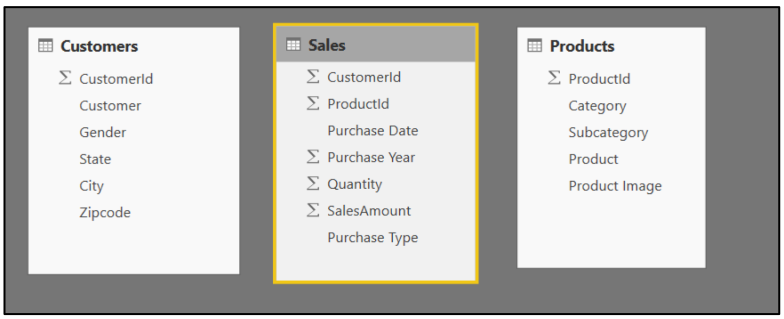
h) In the **Query settings** pane in the right-side of the Query Editor window, change the query name to **Sales**.



i) Click the **Close & Apply** button to close the Query Editor window.

j) Navigate to Relationship View and inspect the set of tables which should now include **Customers, Products** and **Sales**.

k) Use the mouse to rearrange the layout of the three tables to match the following screenshot.



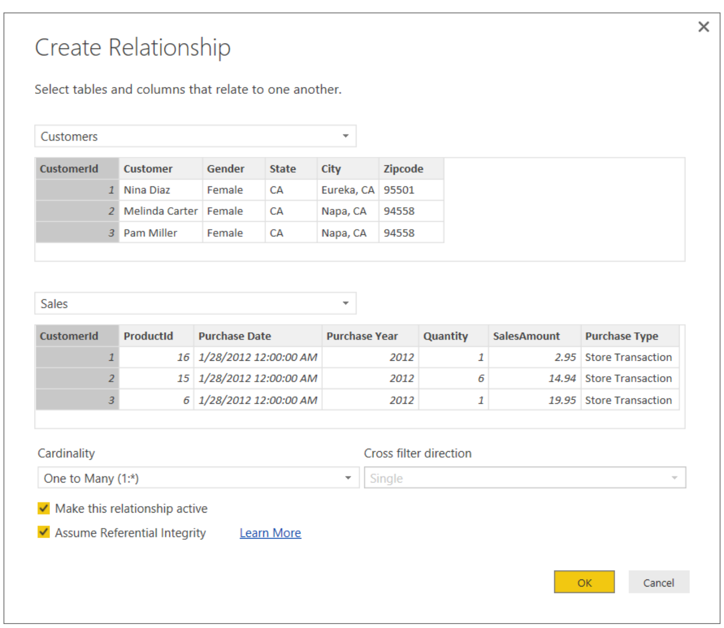
l) Save your work by clicking the **Save** button at the top left of the main Power BI Desktop window.

7. Create a relationship between the **Customers** table and the **Sales** table.

a) In Relationship View, use the mouse the drag and drop the **CustomerId** field from **Customers** table onto the **CustomerId** field from **Sales** table. When you do this, Power BI desktop will prompt you with the **Create Relationship** dialog.

b) In the **Create Relationship** dialog, make sure that the **Make this relationship active** checkbox and the **Assume Referential Integrity** checkbox are both in a checked state.

c) Click the **OK** button to create the new relationship and to dismiss the **Create Relationship** dialog.

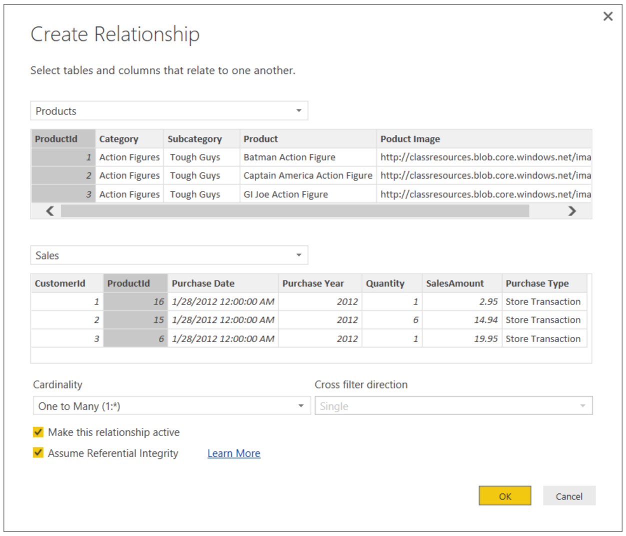


8. Create a relationship between the **Products** table and the **Sales** table.

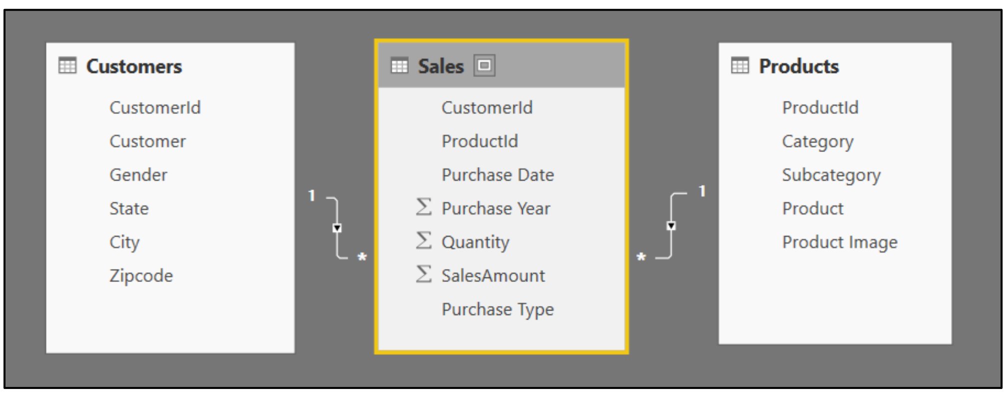
   a) In Relationship View, use the mouse the drag and drop the **ProductId** field from **Products** table onto the **ProductId** field from **Sales** table. When you do this, Power BI desktop will prompt you with the **Create Relationship** dialog.

   b) In the **Create Relationship** dialog, make sure that the **Make this relationship active** checkbox and the **Assume Referential Integrity** checkbox are both in a checked state.

   c) Click the **OK** button to create the new relationship and to dismiss the **Create Relationship** dialog.



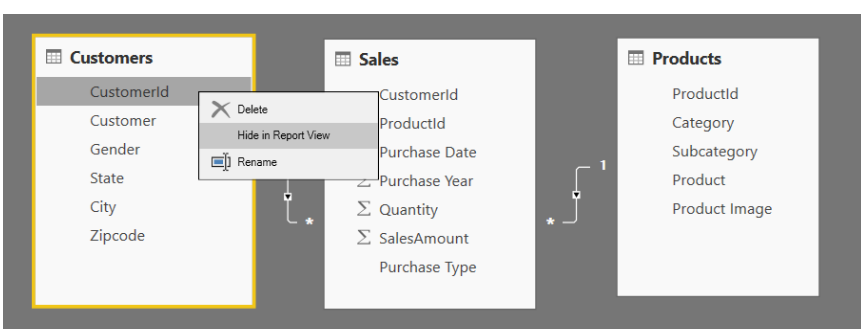
   d) There should now be two tables relationships that relate all three tables together into a single data model.



   e) Save your work by clicking the **Save** button at the top left of the main Power BI Desktop window.

9. Hide the fields in the data model from Report View that will not be used when designing report pages.

   a) In Relationship View, right-click the **CustomerId** field in the Customers table and click the **Hide in Report View** command.



b) Use the same technique to hide the **CustomerId** field in the **Sales** table.

c) Use the same technique to hide the **ProductId** field in the **Sales** table.

d) Use the same technique to hide the **Quantity** field in the **Sales** table.

e) Use the same technique to hide the **SalesAmount** field in the **Sales** table.

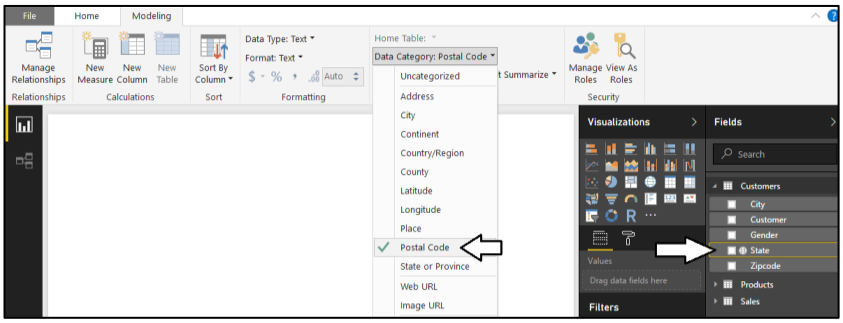
f) Use the same technique to hide the **ProductId** field in the **Products** table.

10. Configure Geolocation Metadata for the **State** field in the **Customers** table.

a) Make sure you are still in Report View.

b) From the **Fields** list on the right, select the **State** field from the **Customers** table.

c) Drop down the **Data Category** menu from the ribbon and select **State or Province**.

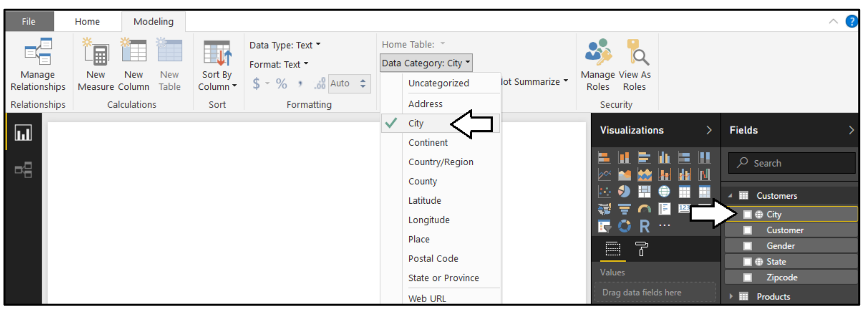


11. Configure geolocation metadata for the **City** field in the **Customers** table.

a) Make sure you are still in Report View.

b) From the **Fields** list on the right, select the **City** field from the **Customers** table.

c) Drop down the **Data Category** menu from the ribbon and select **City**.

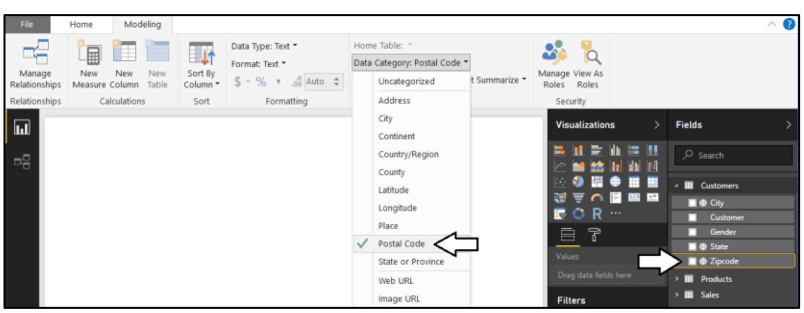


12. Configure geolocation metadata for the **Zipcode** field in the **Customers** table.

a) Make sure you are still in Report View.

b) From the **Fields** list on the right, select the **Zipcode** field from the **Customers** table.

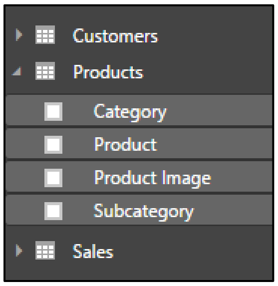
c) Drop down the **Data Category** menu from the ribbon and select **Postal Code**.



d) Save your work by clicking the **Save** button at the top left of the main Power BI Desktop window.

13. Add a new dimensional hierarchy to the **Products** table.

a) Inspect the **Products** table in the **Fields** list. This table should display the fields shown in the following screenshot.



b) Right-click on the **Category** field and then select the **New Hierarchy** menu command.

c) You should now see a new dimensional hierarchy in the fields list named **Category Hierarchy**.

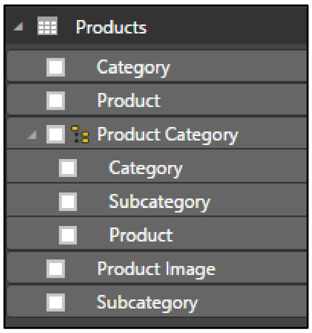
d) Right-click **Category Hierarchy** and select the **Rename** menu command.

e) Rename the new hierarchy **Product Category**.

f) Right-click on the **Subcategory** field and select the **Add to Product Category** menu command.

g) Right-click on the **Product** field and select the **Add to Product Category** menu command.

h) The **Product Category** hierarchy should now contain three fields as shown in the following screenshot.

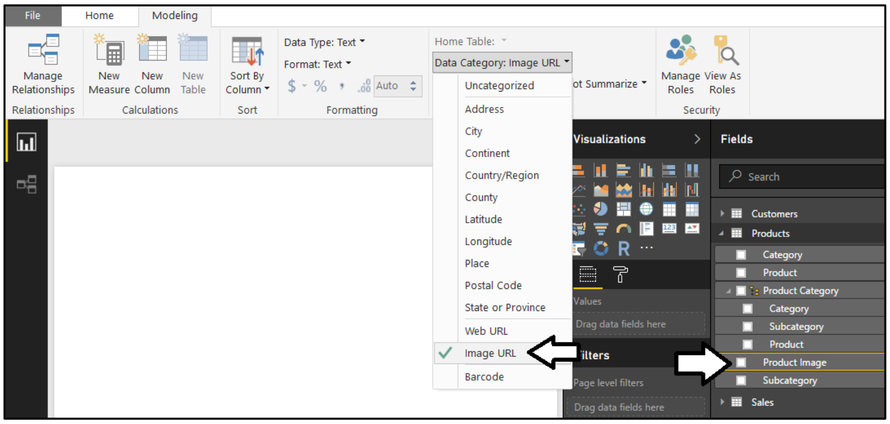


14. Configure image URL metadata for the **Product Image** field in the **Products** table.

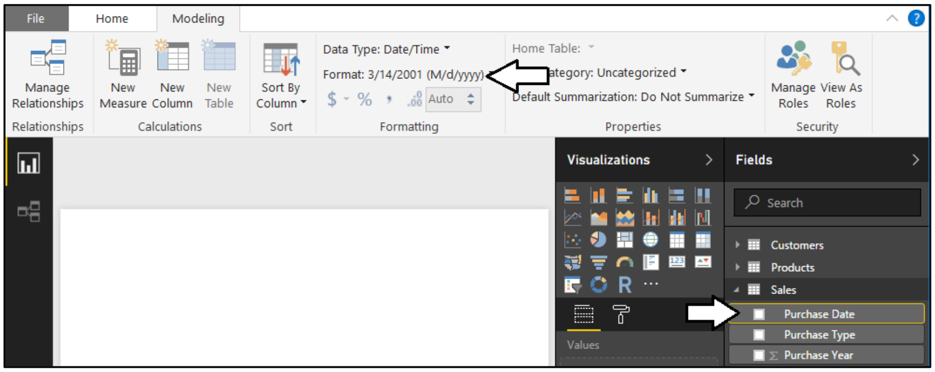
a) Make sure you are still in Report View.

b) From the **Fields** list on the right, select the **Product Image** field from the **Products** table.

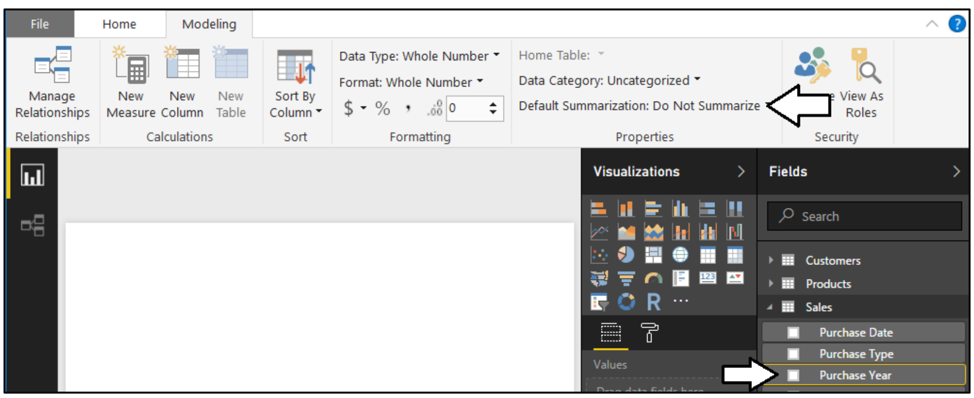
c) Drop down the **Data Category** menu from the ribbon and select **Image URL**.



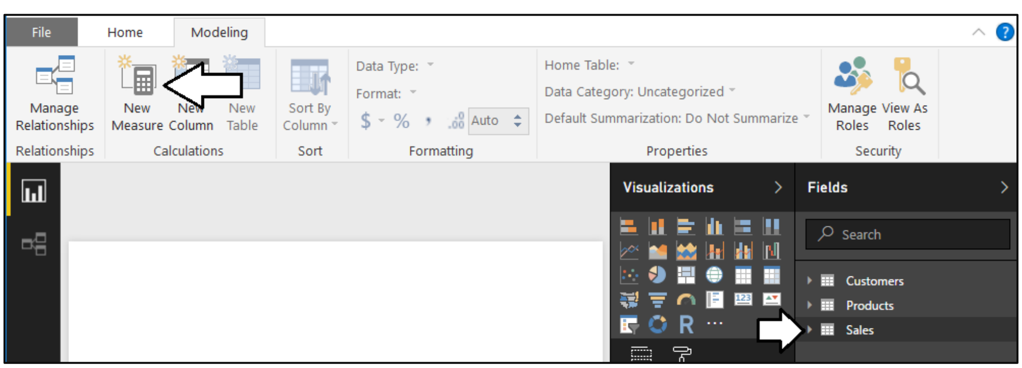
15. Configure the formatting for the **Purchase Date** field in the **Sales** table.
    a) Make sure you are still in Report View.
    b) In the **Fields** list in the right, select the **Purchase Date** field from the **Sales** table.
    c) Set the **Format** property to **3/14/2001 (M/d/yyyy)**.



16. Configure the default summarization for the **Purchase Year** field in the **Sales** table.
    a) Make sure you are still in Report View.
    b) In the **Fields** list in the right, select the **Purchase Year** field from the **Sales** table.
    c) Set the **Default Summarization** property to **Do Not Summarize**.



17. Create a measure named **Units Sold** to perform a sum aggregation on the **Quantity** column of the **Sales** table.
    a) Make sure you are in Report View.
    b) Select the **Sales** table from the **Fields** list.
    c) Create a new measure by clicking the **New Measure** button in the **Modeling** tab of the ribbon.

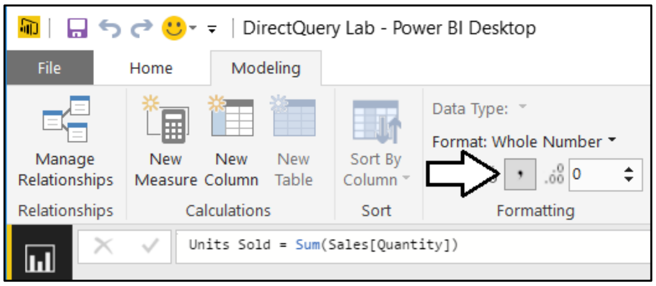


d) Enter to following DAX expression into the formula bar to create the measure named **Units Sold**.

```
Units Sold = SUM(Sales[Quantity])
```

e) Press the **ENTER** key to add the measure to data model.

f) Modify the formatting by clicking and selecting the Comma button on the ribbon to add a comma separator.



18. Create a measure named **Sales Revenue** to perform a sum aggregation on the **SalesAmount** column of the **Sales** table.

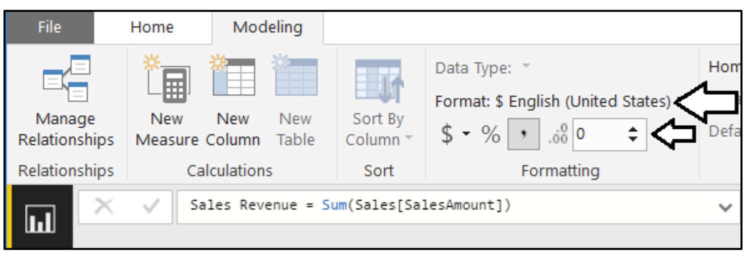
a) Create a new measure by clicking the **New Measure** button in the ribbon.

b) Enter to following DAX expression into the formula bar to create the measure named **Sales Revenue**.

```
Sales Revenue = Sum(Sales[SalesAmount])
```

c) Press the **ENTER** key to add the measure to data model.

d) Modify the formatting by dropping down the **Format** menu on the ribbon and selecting **Currency > $ English (United States)**. Also use the spinner control below the **Format** menu to set the number of decimal places shown to zero.



19. Create a measure named **Customer Count** to perform an aggregation to count the number of rows in the **Customers** table.

a) Make sure the **Customers** table is selected.

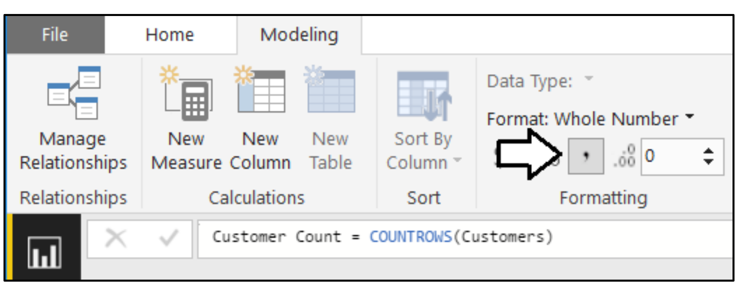
b) Create a new measure by clicking the **New Measure** button in the ribbon.

c) Enter the following DAX expression into the formula bar to create the measure named **Customer Count**.

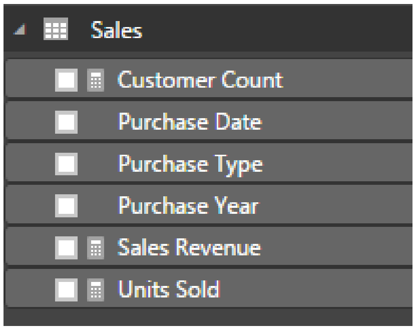
```
Customer Count = COUNTROWS(Customers)
```

d) Press the **ENTER** key to add the measure to data model.

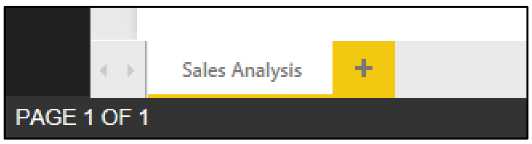
e) Modify the formatting by clicking and selecting the Comma button on the ribbon to add a comma separator.



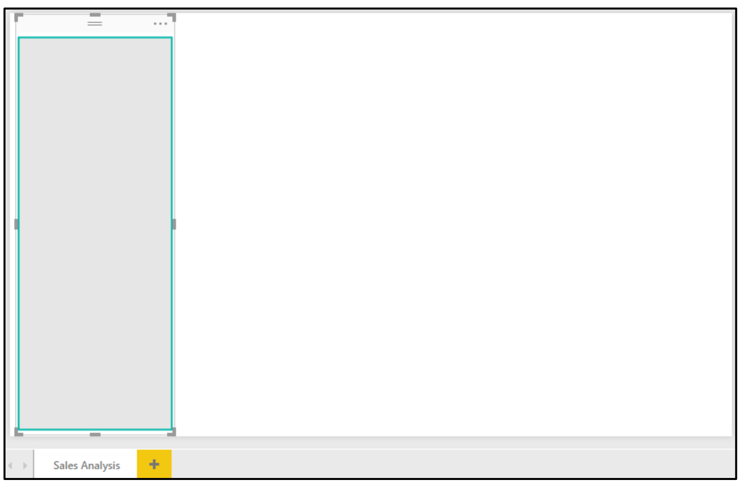
f) At this point, the fields in the Sales table should match the following screenshot.



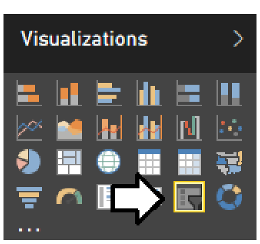
20. Save your work by clicking the **Save** button at the top left of the main Power BI Desktop window.
21. Currently, the report for the current project has a single page. Rename this report page from **Page1** to **Sales Analysis**.



22. Create a rectangle shape on the report page to provide background formatting for a slicer visual.
    a) Drop down the **Shapes** menu and select the **Rectangle** command to add a new shape to the report.
    b) Using the mouse, resize the rectangle shape to take up the full height of the report page and about 20% of the width.

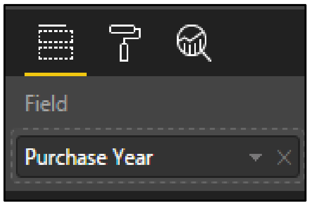


23. Add a new slicer visual to the page to filter by **Purchase Year**.
    a) Click the **New Visual** button on the ribbon to add a new visual to the page.
    b) Change the visual to a slicer by clicking the **Slicer** button in the **Visualizations** list.

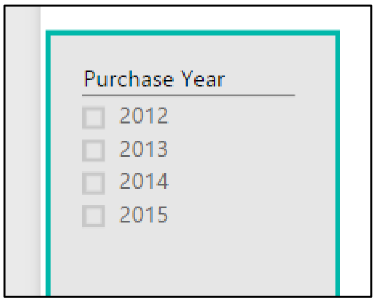


c) Position the slicer on top of the rectangle.

d) Drag and drop the **Purchase Year** column from the **Sales** table into the **Values** well.



e) You should now have a slicer visual that matches the following screenshot.



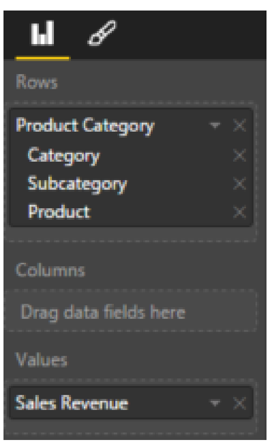
24. Create a new matrix visual to display sales revenue broken out by product category, subcategory and product.

a) Click the **New Visual** button on the ribbon to add a new visual to the page.

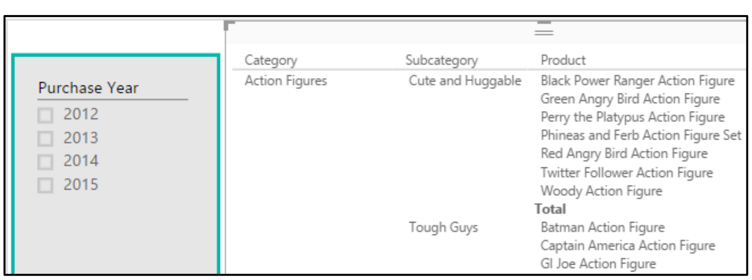
b) Change the visual to a matrix by clicking the **Matrix** button in the **Visualizations** list.

c) In the **Fields** list, click the checkbox for the **Product Category**.

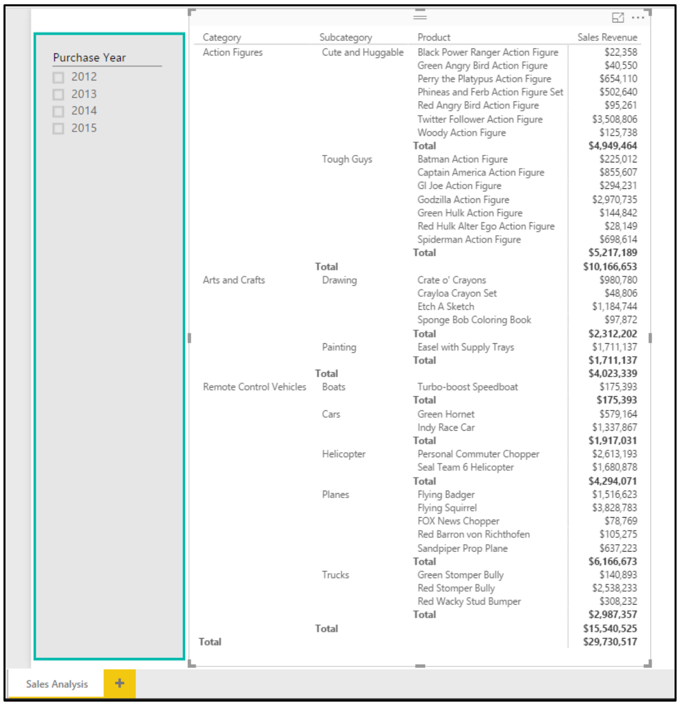
d) Drag and drop the **Sales Revenue** field from the **Sales** table into the **Values** well.



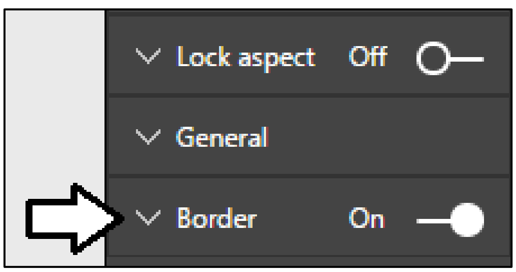
e) The Matrix visual should now display sales revenue for each product along with totals for each subcategory and category.



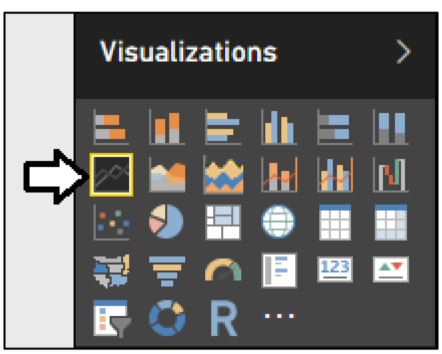
f) Reposition the matrix visual so it displays all its columns and so that it takes up the entire height of the report page.



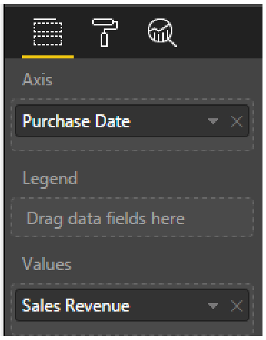
g) Configure the **Border** property of the matrix visual to **On**.



25. Save your work by clicking the **Save** button at the top left of the main Power BI Desktop window.
26. Add a new line chart visual to display Sales Revenue by Purchase Date.
    a) Click the **New Visual** button on the ribbon to create a new visual.
    b) Click the **Line chart** button in the **Visualizations** list to change the visual into a Line chart visual.

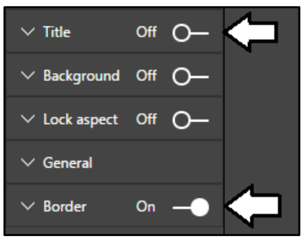


c) Configure the fields for the **Line Chart** visual by adding the **Purchase Date** field from the **Sale** table into the **Axis** well and the **Sales Revenue** field from the **Sales** table into the **Values** well.

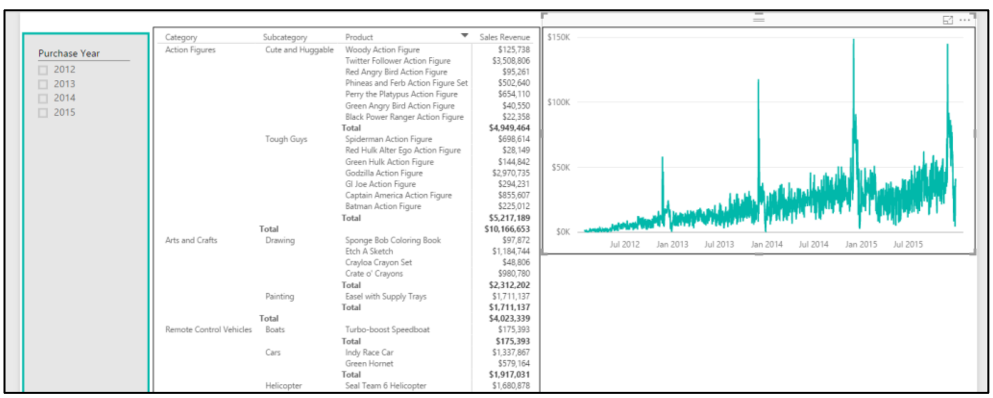


d) Configure the **Title** property of the matrix visual to **Off**.

e) Configure the **Border** property of the matrix visual to **On**.



f) Position the line chart visual in the upper right-hand corner of the report page.

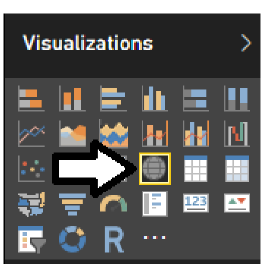


27. Save your work by clicking the **Save** button at the top left of the main Power BI Desktop window.

28. Add a new Map visual to display sales revenue by state.

a) Click the **New Visual** button on the ribbon to create a new visual.

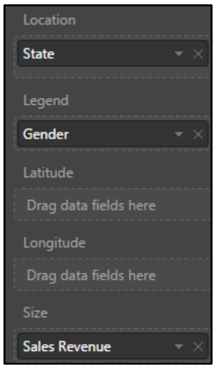
b) Click the **Map** button in the **Visualizations** list to change the visual into a Map visual.



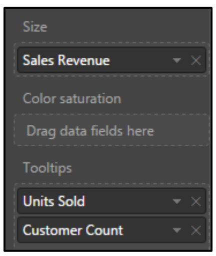
c) Configure the fields for the **Map** visual by adding the **State** field from the **Customers** table into the **Location** well.

d) Add the **Gender** field from the **Customers** table into the **Legend** well.

e) Add the **Sales Revenue** field from the **Sales** table into the **Size** well.

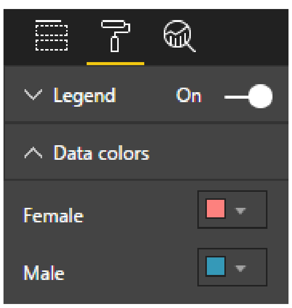


f) Add the **Units Sold** field and the **Customer Count** field from the **Sales** table into the **Tooltips** well.



g) With the Map visual still selected, switch over the Format properties pane.

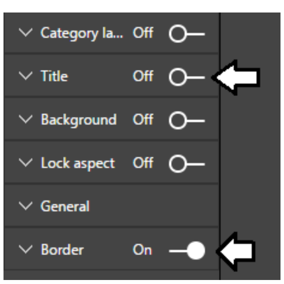
h) In the **Data colors** section, change the **Female** color to pink and the **Male** color to blue.



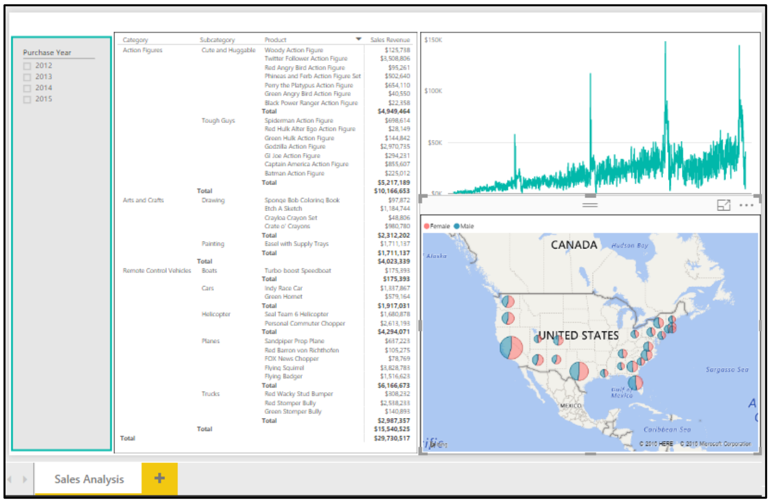
Alternatively, you might prefer to select two colors other than pink and blue that don't perpetuate age-old, gender-based stereotypes.

i) Configure the **Title** property of the matrix visual to **Off**.

j) Configure the **Border** property of the matrix visual to **On**.



k) Reposition the map visual so it appears in the bottom left corner of the report page.

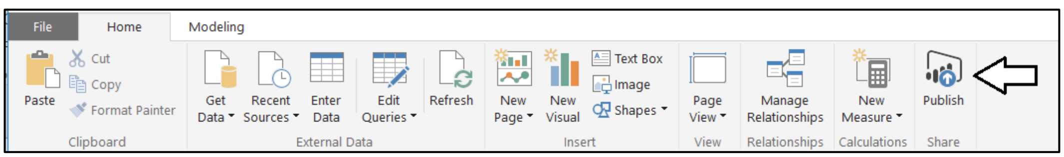


29. Save your work by clicking the **Save** button at the top left of the main Power BI Desktop window.

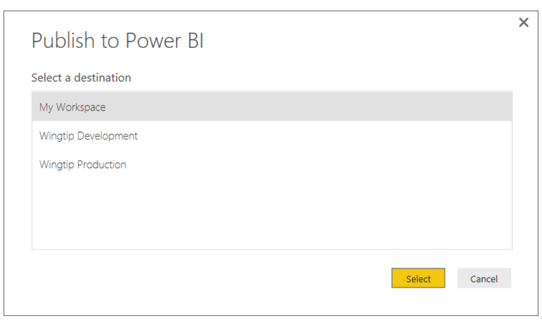
30. Publish the project to the Power BI service.

a) Navigate to **Home** tab in ribbon

b) Click the **Publish** button on the far right-hand side of the ribbon.

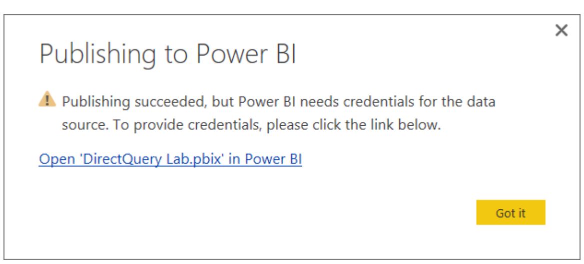


c) When prompted to **Select a destination**, choose **My Workspace** then click the **Select** button.



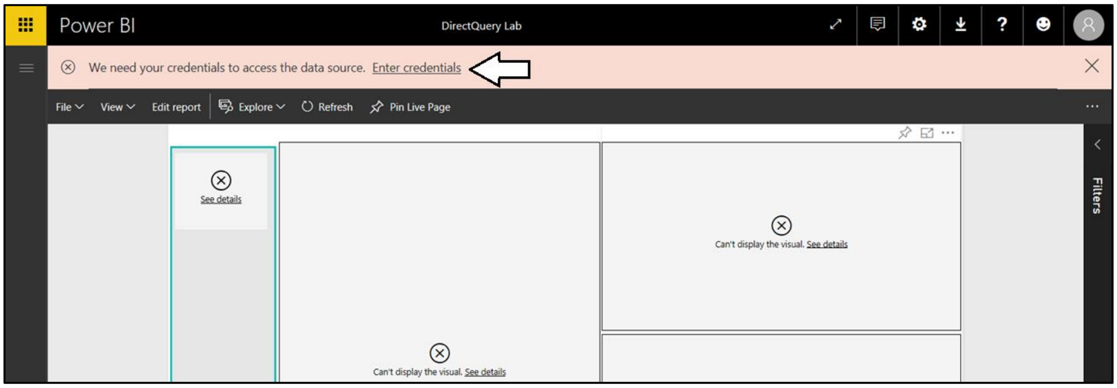
d) After a project which uses DirectQuery mode has been published, Power BI Desktop will prompt you with a dialog which informs you that you must configure the data source credentials before the report can be used.

e) Click the link with the caption **Open 'DirectQuery Lab' in Power BI** to navigate to the Power BI service in the browser.



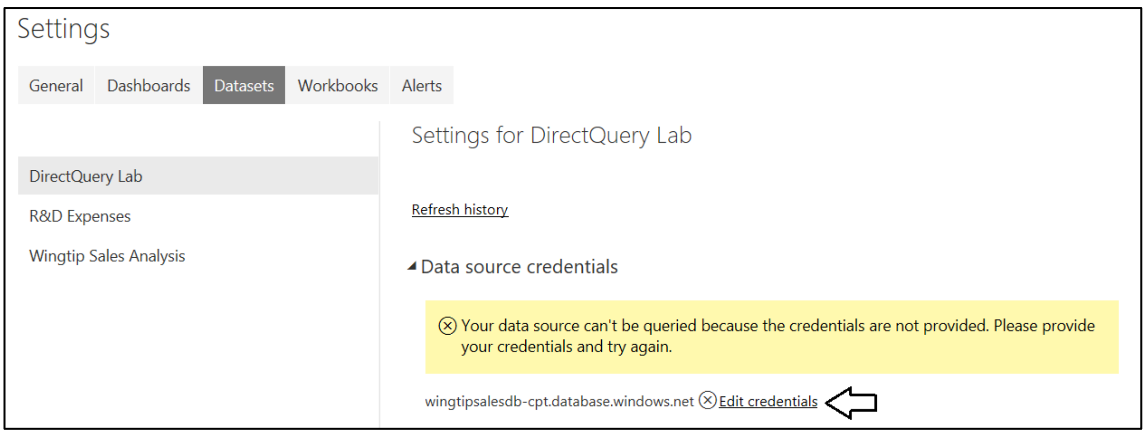
f) When you navigate to the report for the **DirectQuery Lab** project, you will notice that the visuals cannot render their data and there is a message bar with the error message **"We need your credentials to access the data source"**.

g) Click the **Enter credentials** link to the right of the error message to redirect to the data source settings page.



h) When you inspect the **Data source credentials** section for the **DirectQuery Lab** dataset, you should see another error message indicating the data source cannot be queried because no credentials have been provided.

i) Click the **Edit credentials** link.

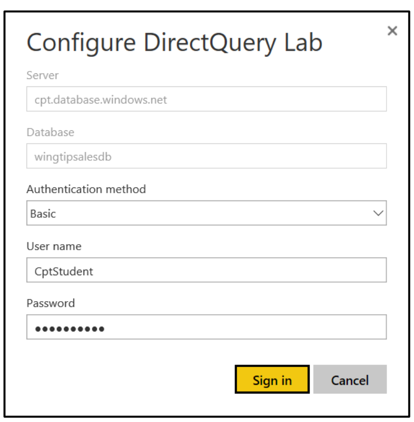


j) On the **Configure DirectQuery Lab** dialog, set the **Authentication method** property to **Basic**

k) Enter a **User name** of **CptStudent**.

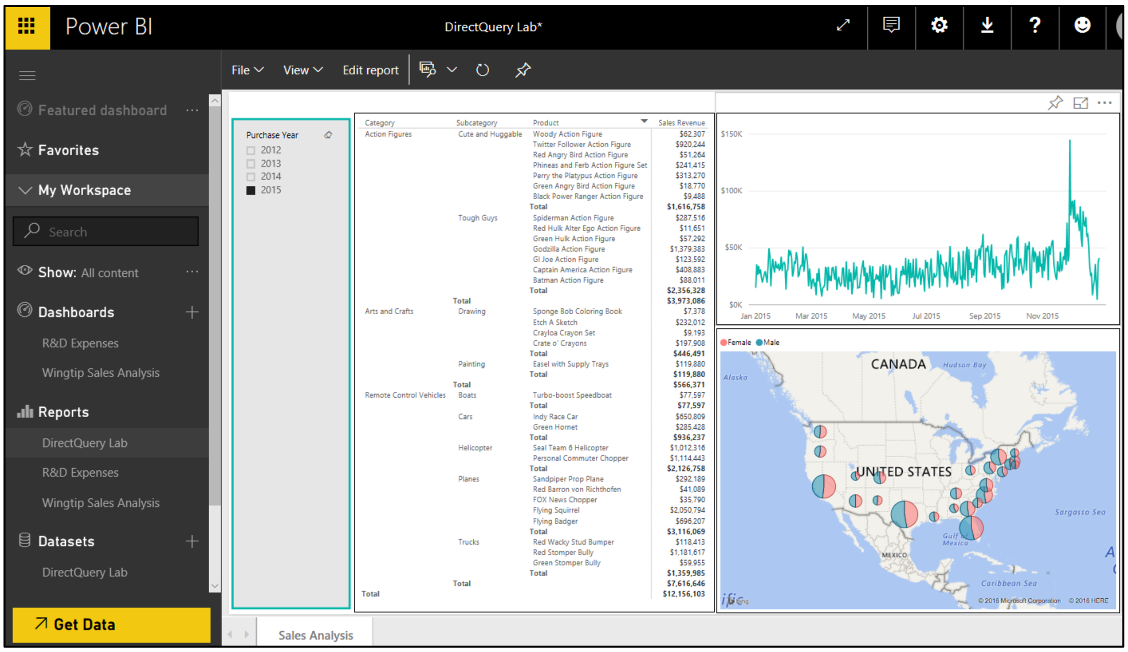
l) Enter a **Password** of **pass@word1**.

m) Click the **Sign In** button to save the credentials and cache them inside the Power BI service.



n) The Power BI service should display a notification that the data source has been updated.

o) At this point, you should now be able to access the **DirectQuery Lab** report. Each time you select a different year in the slicer, the Power BI service will execute a query against the Azure SQL Server database to update the report.



Congratulations. You have now completed this lab exercise.