

Using the Power Query Features of Power BI Desktop

Lab Time: 40 minutes

Lab Folder: C:\Student\Modules\PowerQuery\Lab\

Lab Overview: In this lab you will begin by creating a new Power BI Desktop project and saving it as a PBIX file. Next, you will learn to work with the Power Query features of Power BI Desktop to extract data from a SQL Azure database and to transform the data as it is loaded into the data model. This is the first lab in a sequence of labs that continue with the same PBIX file. In other words, the labs that follow will build upon the work you do in this lab.

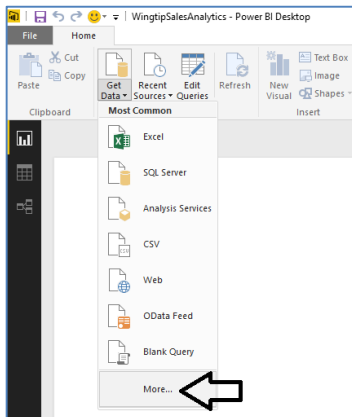
Exercise 1: Importing Data from a SQL Azure Database

In this exercise you will create and save a new Power BI Desktop project. After that, you will connect to a SQL Azure database and import data into Power BI Desktop using its Power Query features.

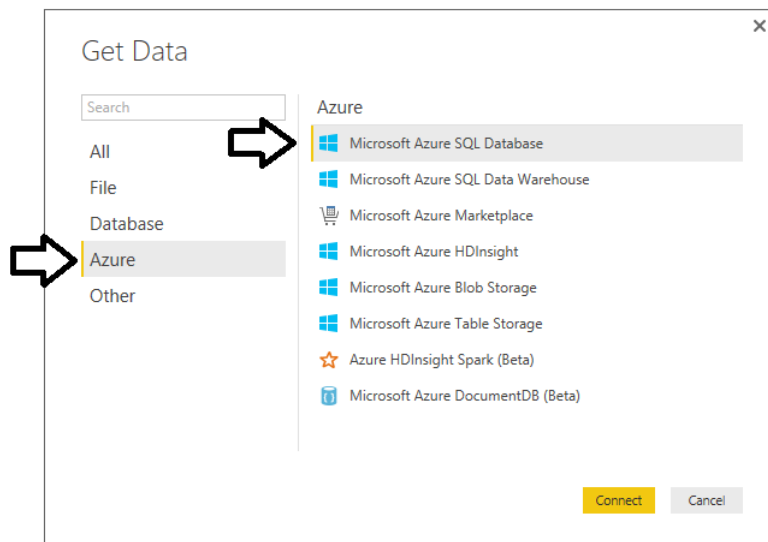
1. Launch Power BI Desktop to start a new project.
2. Save the new project as **WingtipSalesAnalytics.pbix** using the following path.

C:\Student\Projects\WingtipSalesAnalytics.pbix

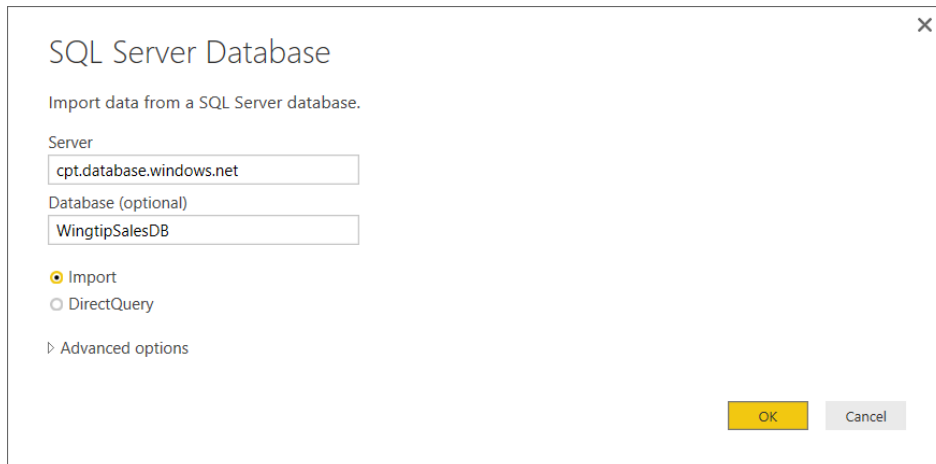
3. Drop down the **Get Data** menu button on the ribbon and click **More....**



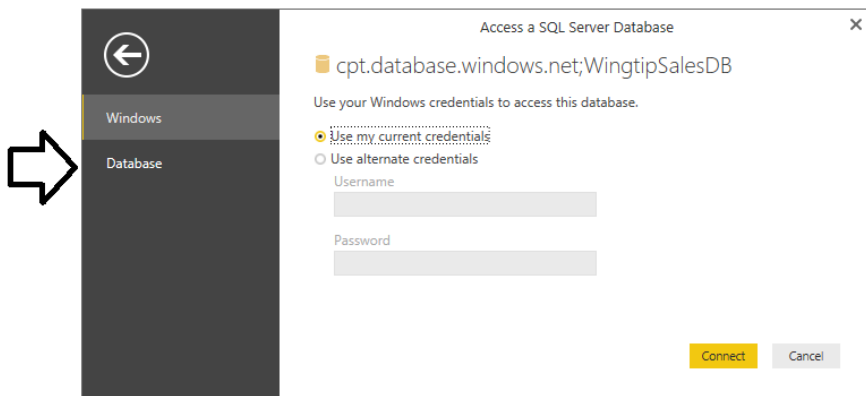
4. On the **Get Data** dialog, select **Azure** in the list on the left. Next, select **Microsoft SQL Azure Database** on the right and then click the **Connect** button.



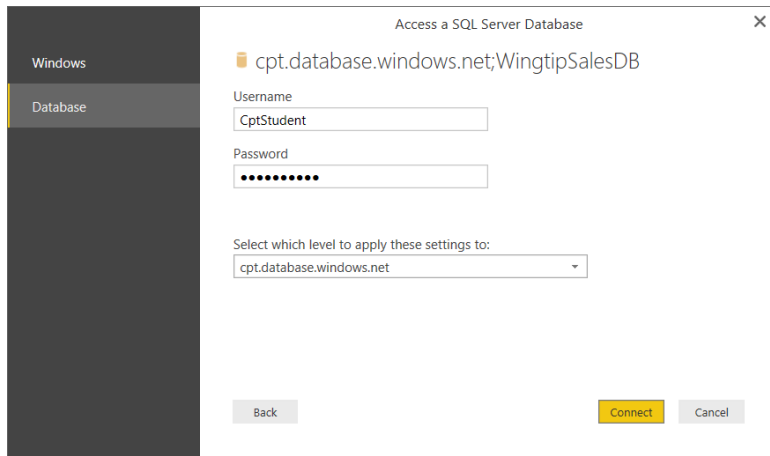
5. When you are prompted with the **SQL Server Database** dialog, complete the following tasks.
- Enter a **Server** value of **cpt.database.windows.net**
 - Enter a **Database** value of **WingtipSalesDB**
 - Leave the option button with the default setting of **Import** and not **DirectQuery**.
 - Click the **OK** button to continue.



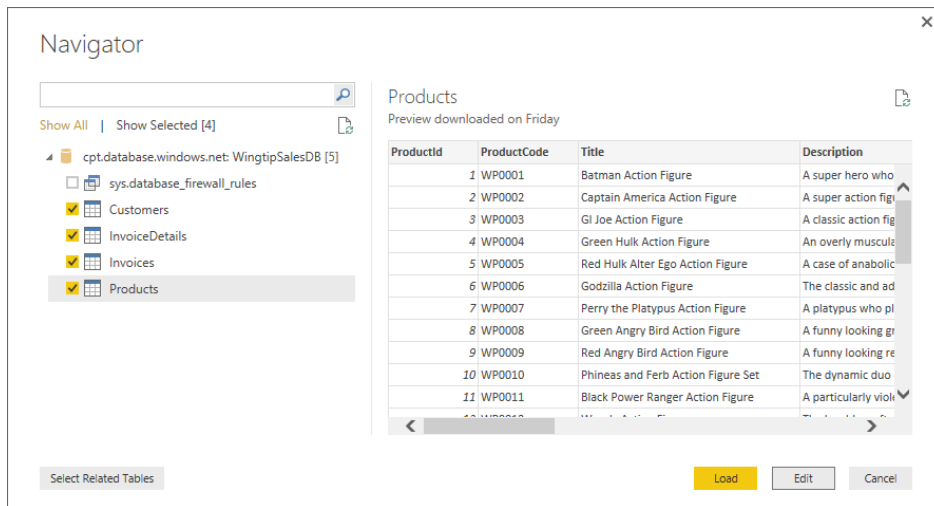
6. At this point, you will be prompted by the **Access a SQL Server Database** dialog. Click on **Database** on the left side of the dialog so that you can enter the credentials for a standard SQL account instead of using Windows authentication.



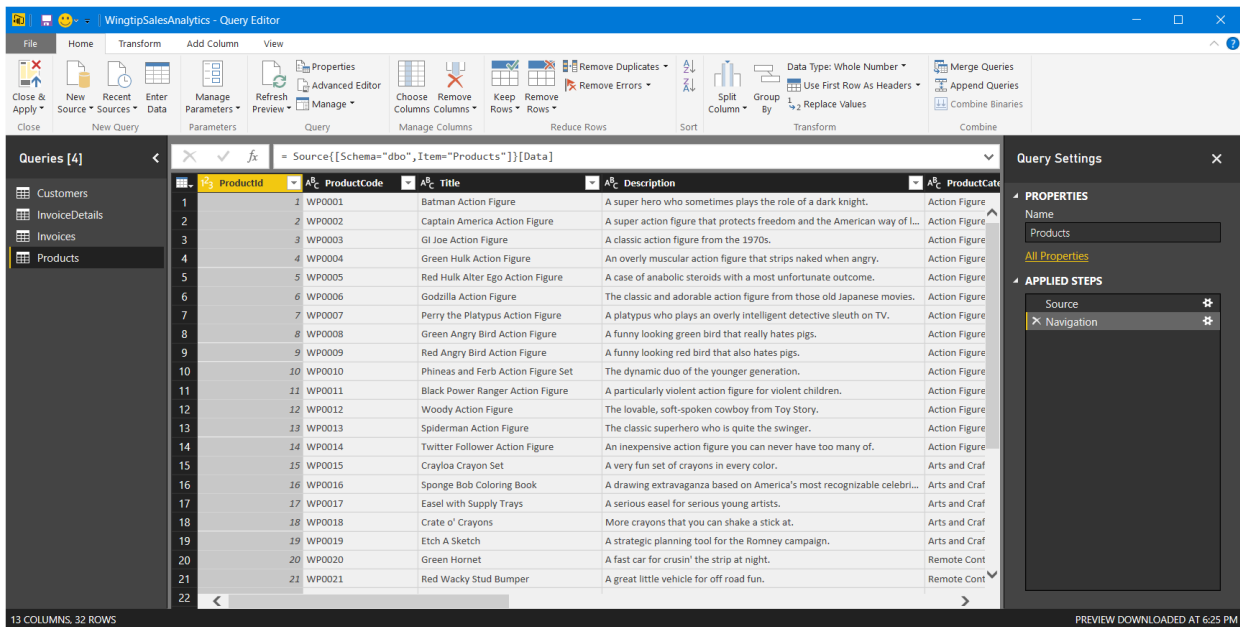
7. Enter the following credentials for a SQL user account that has been configured with read access to the database.
- Username: **CptStudent**
 - Password: **pass@word1**
8. Once you have entered the credentials the **Access a SQL Server Database** dialog, click the **Connect** button to continue.



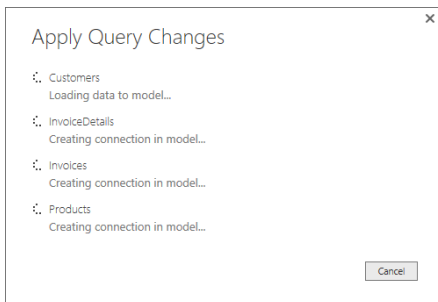
9. At this point, Power BI Desktop should be able to establish a connection to the database and then prompt you with the **Navigator** dialog. The **Navigator** dialog allows you to select the tables you would like to import into your PBIX project.
10. In the **Navigator** dialog, select the **Customers** table, the **InvoiceDetails** table, the **Invoices** table and the **Products** table as shown in the following screenshot. Once you have selected these four tables, click the **Edit** button to create a query for each of these tables and to open the **Query Editor** dialog.



11. When you inspect the **Query Editor** window, you should now be able to observe that Power BI Desktop has created a new query for each of the four tables that you selected in the **Navigator** dialog. You will not begin to edit these queries until the next exercise. However, now you should take a moment to inspect each query by clicking on the query name in the **Queries** list on the left. You should be able to observe that each query is initially created to return the data just as it is defined in the underlying table.



- Click the **Close and Apply** button in the **Query Editor** window. This will close the **Query Editor** window and start the process of extracting the data from the SQL Azure database and importing it into the current project's data model using a local cache on your student workstation. The **Apply Query Changes** dialog will be displayed with spinning icons as Power BI Desktop imports that data.

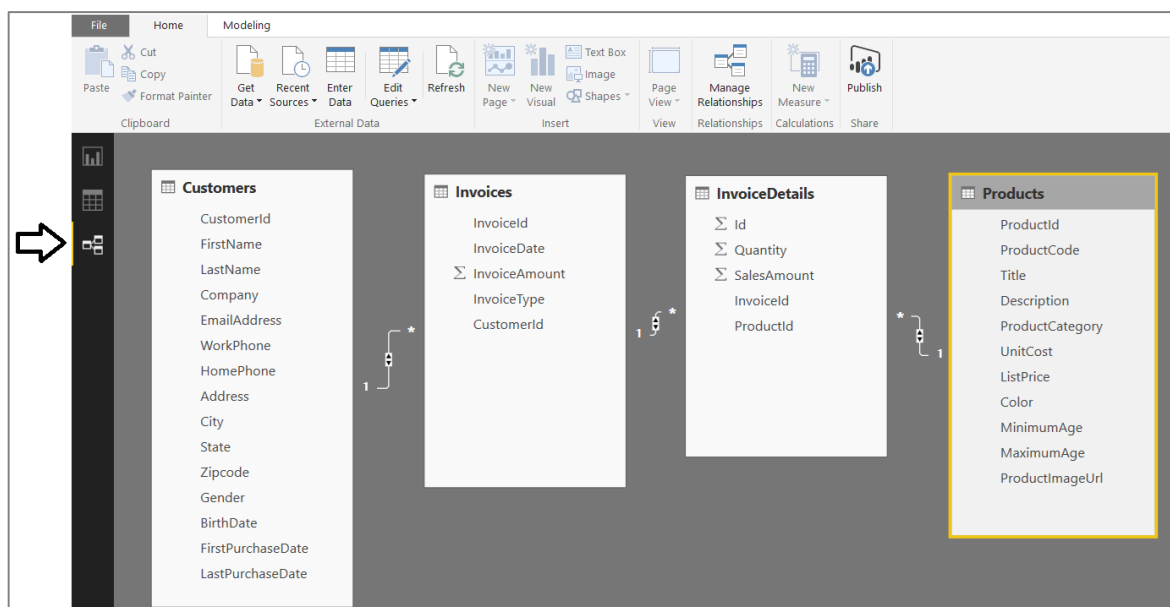


- Once the data from all four queries has been imported, examine the **Fields** list on the right-hand side of the Power BI Desktop window. You should be able to see that four tables have been created from the four queries.

CustomerId	FirstName	LastName	Company	EmailAddress
760	Lucile	Blake	Shinra Electric Power Companyworld, Final Fantasy VII	Lucile.Blake@ShinraElectric
881	Rochelle	Owen	Peach Pit	Rochelle.Owen@PeachPit.c
940	Corinne	Finch	North Western Railway	Corinne.Finch@NorthWeste
1119	Twila	Massey	Yoyodyne Propulsion Systems	Twila.Massey@YoyodynePr
1520	Colette	Walls	Rovers Return	Colette.Walls@RoversRetur
1548	Kellie	Yang	Uplink Corporation	Kellie.Yang@UplinkCorpora
1552	Felicia	Nash	Duff Beer	Felicia.Nash@DuffBeer.com
1711	Isabel	Ewing	Roxxon	Isabel.Ewing@Roxxon.com
2195	Megan	Martin	Slate Rock and Gravel Company	Megan.Martin@SlateRocka
2252	Cynthia	Blake	Sirius Cybernetics Corporation	Cynthia.Blake@SiriusCyber
2341	Karyn	Hodges	Daystrom Data Concepts	Karyn.Hodges@DaystromDi
2368	Priscilla	Potter	Roxxon	Priscilla.Potter@Roxxon.cor
2525	Nadia	Gray	Grim Reaper Airways	Nadia.Gray@GrimReaperAi
2701	Noemi	Holmes	Medical Mechanica	Noemi.Holmes@MedicalMe
2893	Amanda	Pruitt	Milliways	Amanda.Pruitt@Milliways.c
3007	Teri	Dale	Shinra Electric Power Companyworld, Final Fantasy VII	Teri.Dale@ShinraElectricPoi

TABLE: Customers (63,683 rows)

14. Click on the bottom icon in the sidebar to navigate to relationship view. You should be able to see each of the four tables. You should also be able to see that Power BI Desktop has automatically created relationships between the tables as they were imported into the data model. Use your mouse to move and resize the tables in relationship view so you can see all the fields in each table.

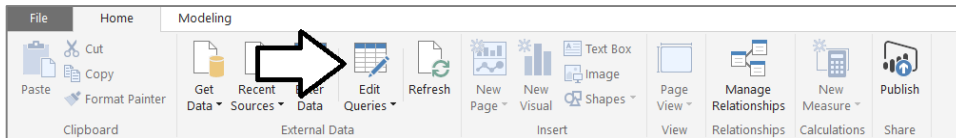


15. Save your work in the current project by clicking the Save button in the upper left corner of the Power BI Desktop window.

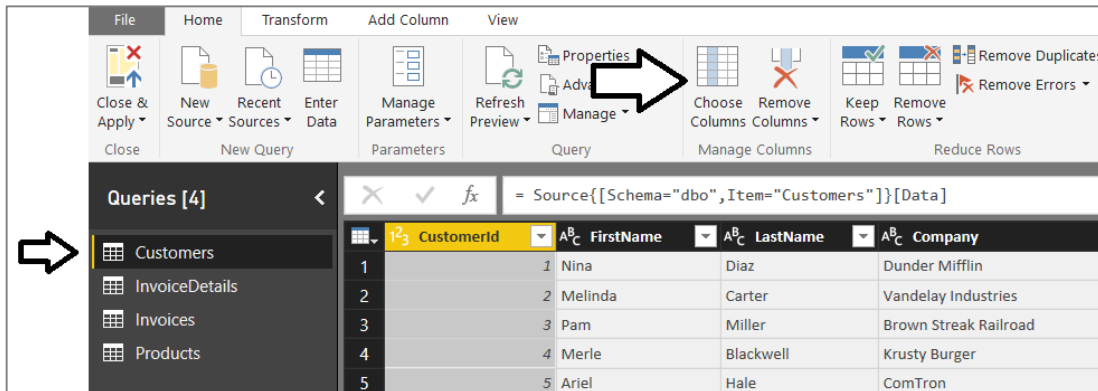
Exercise 2: Using Power Query to Transform and Reshape Customer Data

In the following exercise, you will use the **Query Editor** window to modify the **Customers** query to perform transforms on customer data as it is being loaded into the data model.

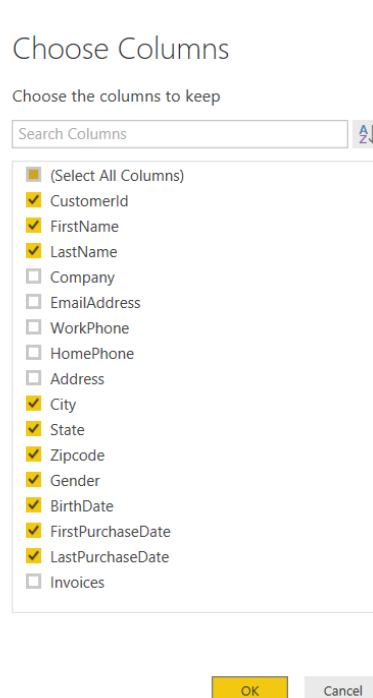
1. Make sure you have the **WingtipSalesAnalytics.pbix** project open that you started in the previous exercise.
2. Click on the **Edit Queries** button in the ribbon to display the Query Editor window.



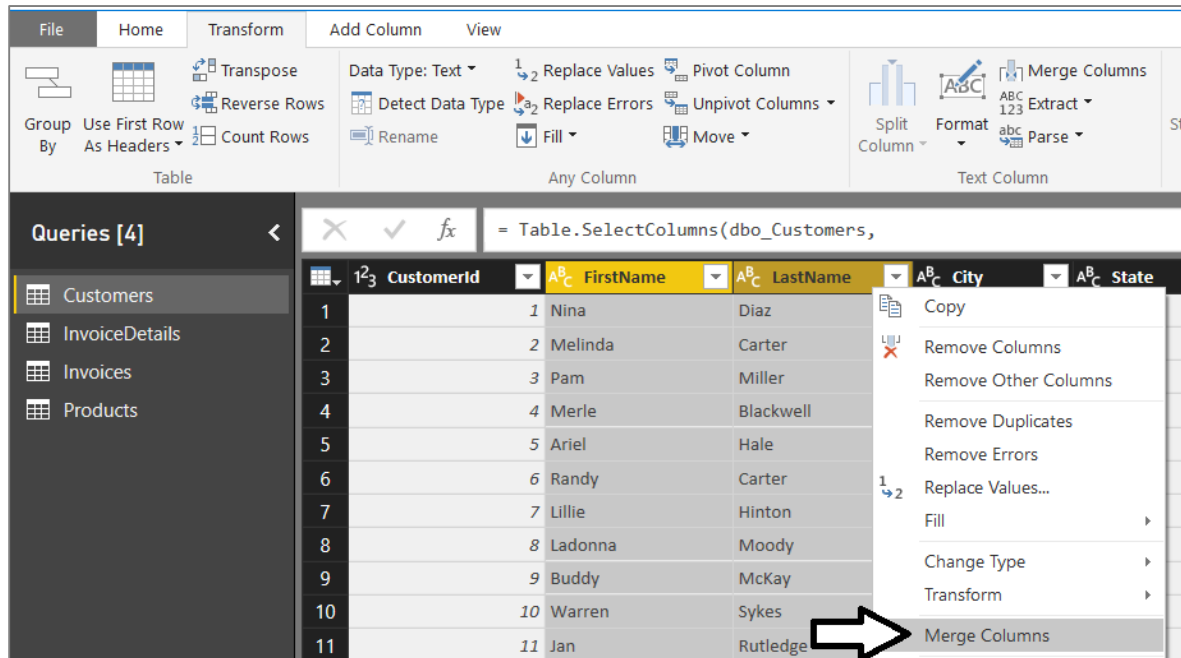
- Make sure the **Customers** query is selected in the **Queries** list on the left-hand side of the Query Editor window. Click the **Choose Columns** button in the ribbon to display the **Choose Columns** dialog.



- In the **Choose Columns** dialog, begin by clicking on the **(Select all Columns)** checkbox at the top to unselect all column. Next, select the checkboxes for **CustomerId**, **FirstName**, **LastName**, **City**, **State**, **Zipcode**, **Gender**, **BirthDate**, **FirstPurchaseDate** and **LastPurchaseDate** as shown in the following screenshot. Once you have these columns selected, click the **OK** button to close the **Choose Columns** dialog and to modify the underlying query.



- You should be able to see that the Query Editor window now only shows the columns that you selected.
- In this step you will merge the **FirstName** column and the **LastName** column together into a single column named **Customer**.
 - Select the **FirstName** column by clicking on its column header.
 - Next, hold down the **SHIFT** key and select the **LastName** column by clicking on its column header.
 - Right-click on the selected columns and click the **Merge Columns** menu command.



- d) In the **Merge Column** dialog, drop down the **Separator** control and select a value of **Space**. Add a **New column name** value of **Customer** and click the **OK** button to modify the underlying query with your changes.

The 'Merge Columns' dialog box is shown. It has a title bar with a close button (X). The text 'Choose how to merge the selected columns.' is at the top. Below it, the 'Separator' dropdown menu is set to 'Space'. The 'New column name (optional)' text box contains the text 'Customer'. At the bottom right, there are 'OK' and 'Cancel' buttons. The 'OK' button is highlighted in yellow.

- e) You should now be able to see that the **FirstName** column and the **LastName** column have been replaced with a single merged column named **Customer**.

	1 ² ₃	CustomerId	AB _C Customer	AB _C City	AB _C State
1		1	Nina Diaz	Eureka	CA
2		2	Melinda Carter	Napa	CA
3		3	Pam Miller	Napa	CA
4		4	Merle Blackwell	Sacramento	CA
5		5	Ariel Hale	Sacramento	CA
6		6	Randy Carter	Sacramento	CA

7. The **State** column is currently to the right of the **City** column. Move the **State** column so that it is repositioned to the left of the **City** column. Accomplish this by clicking on the column header for the **State** column and leaving the mouse button down. Drag the **State** column to the left of the **City** column and release the mouse button.

	1 ² 3 CustomerId	A ^B C Customer	A ^B C State	A ^B C City
1		1 Nina Diaz	CA	Eureka
2		2 Melinda Carter	CA	Napa
3		3 Pam Miller	CA	Napa
4		4 Merle Blackwell	CA	Sacramento
5		5 Ariel Hale	CA	Sacramento
6		6 Randy Carter	CA	Sacramento

8. Modify the query so that the **Gender** column returns values of **Male** and **Female** instead of **M** and **F**.
 - a) Make sure the **Transform** tab is the active tab in the ribbon.
 - b) Select the **Gender** column by clicking its column header.
 - c) Click the **Replace Values** button in the ribbon to display the **Replace Values** dialog.

The screenshot shows the 'WingtipSalesAnalytics - Query Editor' interface. The 'Transform' tab is active in the ribbon. The 'Replace Values' button is highlighted with a black arrow. Below the ribbon, the query editor shows a table with columns: CustomerId, Customer, State, City, Zipcode, Gender, and BirthDate. The 'Gender' column is selected, and its values are 'F' and 'M'.

- d) In the **Replace Value** dialog, enter a value of **F** in the **Value to Find** textbox and enter a value of **Female** in the **Replace With** textbox. Click to **OK** button add your changes to the underlying query.

Replace Values

Replace one value with another in the selected columns.

Value To Find

F

Replace With

Female

Advanced options

OK

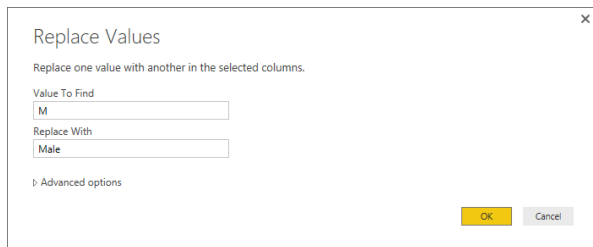
Cancel

- e) You should be able to see that all values of **F** in the **Gender** column have been replaced with a value of **Female**.

A ^B C Gender
Female
Female
Female
Female
M
M
Female
Female
M

- f) Make sure the **Gender** column is still selected.
- g) Click the **Replace Values** button in the ribbon to display the **Replace Values** dialog.

- h) In the **Replace Value** dialog, enter a value of **M** in the **Value to Find** textbox and enter a value of **Male** in the **Replace With** textbox. Click to **OK** button add your changes to the underlying query.



Replace Values

Replace one value with another in the selected columns.

Value To Find
M

Replace With
Male

Advanced options

OK Cancel

- i) You should be able to confirm that all values in the **Gender** column have been replaced with a value of either **Male** or **Female**. If you inspect the **APPLIED STEPS** list in the **Query Settings** pane, you should be able to see that there are two steps at the end that have been given the generic names of **Replaced Value** and **Replaced Value 1**.

Gender	BirthDate	FirstPurchase
Female	4/11/1966 12:00:00 AM	1/28/2
Female	6/6/1976 12:00:00 AM	1/28/2
Female	9/8/1952 12:00:00 AM	1/28/2
Female	9/12/1939 12:00:00 AM	1/28/2
Male	9/15/1965 12:00:00 AM	1/28/2
Male	7/14/1953 12:00:00 AM	1/28/2
Female	2/3/1992 12:00:00 AM	1/28/2
Female	4/5/1949 12:00:00 AM	1/28/2
Male	5/10/1989 12:00:00 AM	1/28/2
Male	6/17/1960 12:00:00 AM	1/28/2
Female	11/26/1981 12:00:00 AM	1/30/2
Male	3/26/1973 12:00:00 AM	1/30/2
Male	4/5/1988 12:00:00 AM	1/30/2

PROPERTIES

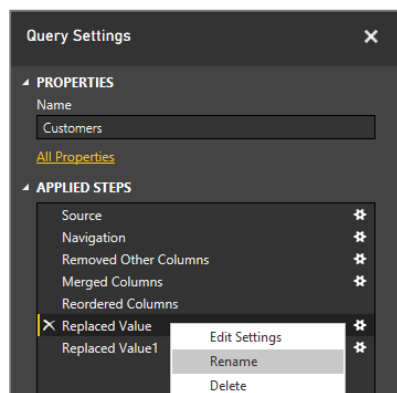
Name
Customers

[All Properties](#)

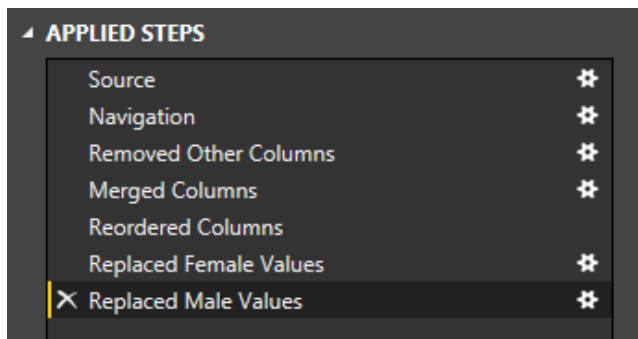
APPLIED STEPS

- Source
- Navigation
- Removed Other Columns
- Merged Columns
- Reordered Columns
- Replaced Value
- Replaced Value 1

- j) In order to promote higher levels of maintainability, it's often a good idea to rename steps with names such as of **Replaced Value** and **Replaced Value 1**. Rename the **Replaced Values** step by right-clicking it and clicking the **Rename** command to place the step name in edit mode. Modify the name of this step to **Replaced Female Values**.



- k) Using the same technique, rename the **Replaced Value 1** step to **Replaced Male Values**.



You have now learned how to rename a query step. Note that this lab exercise will not continue to ask you to change the name of every step you add due to time constraints. However, when you are creating queries in larger, real-world projects that involve multiple team members, it might make sense to rename query steps to make your query logic easier for others to read and understand.

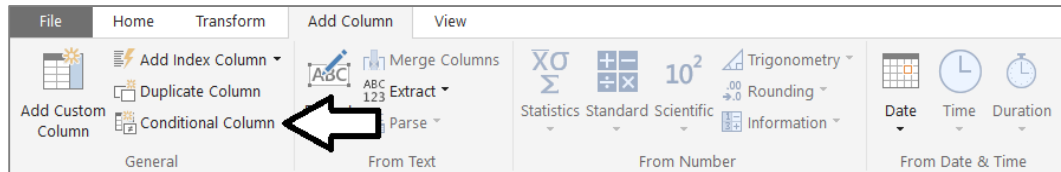
9. Change the column type of **BirthDate**, **FirstPurchaseDate** and **LastPurchaseDate** from **Date/Time** to **Date**.
 - a) Use the column type drop down on the left-hand side of the **BirthDate** column to configure the column using the Date type.

Gender	BirthDate	FirstPurchaseDate
Female	1.2 Decimal Number	1/28/2012 12:00:00 AM
Female	\$ Fixed Decimal Number	1/28/2012 12:00:00 AM
Female	123 Whole Number	1/28/2012 12:00:00 AM
Female	Date/Time	1/28/2012 12:00:00 AM
Male	Date	1/28/2012 12:00:00 AM
Male	Time	1/28/2012 12:00:00 AM
Female	Date/Time/Timezone	1/28/2012 12:00:00 AM
Female	Duration	1/28/2012 12:00:00 AM

- b) Use the column type drop down of the **FirstPurchaseDate** column to configure the column using the Date type.
 - c) Use the column type drop down of the **LastPurchaseDate** column to configure the column using the Date type.
 - d) You should see that the three columns now show values with a date but without a time.

BirthDate	FirstPurchaseDate	LastPurchaseDate
4/11/1966	1/28/2012	1/28/2012
6/6/1976	1/28/2012	1/28/2012
9/8/1952	1/28/2012	1/28/2012
9/12/1939	1/28/2012	1/28/2012
9/15/1965	1/28/2012	1/28/2012
7/14/1953	1/28/2012	1/28/2012
2/3/1992	1/28/2012	1/28/2012

10. Add a new conditional column named **CustomerType** to indicates whether the customer is a repeat customer or not.
 - a) Make sure the Customers query is still selected as the active query.
 - b) Activate the **Add Column** tab in the ribbon.
 - c) Click the **Conditional Column** button in the ribbon to display the **Add Custom Column** dialog.



In this particular scenario, you are working under the assumption that the customer is a repeat customer when the **FirstPurchaseDate** column and the **LastPurchaseDate** column are not equal indicating the customer has made two or more purchases.

- d) In the **Add Conditional Column** dialog, enter a **New column name** value of **Customer Type**.
- e) Configure a rule to return a string value of "One-time Customer" if **FirstPurchaseDate** equals **LastPurchaseDate**.
- f) For the **Otherwise** evaluation, return a string value of "Repeat Customer".
- g) When the **Add Conditional Column** dialog matches the screenshot below, click the **OK** button to add the new column.

Add Conditional Column

Add a conditional column that is computed from the other columns or values.

New column name

Column Name	Operator	Value	Output
If <input type="text" value="FirstPurchaseDate"/>	<input type="text" value="equals"/>	<input type="text" value="LastPurchaseDate"/>	Then <input type="text" value="One-time Customer"/>

+ Add Rule

Otherwise

OK Cancel

- h) You should be able to verify that the new **Customer Type** column has a value of **Repeat Customer** when the current customer has a **FirstPurchaseDate** column value that is not equal to the **LastPurchaseDate** column value. When these column values are equal, the **CustomerType** column has a value of **One-time Customer**.

You might have to scroll down several pages of records in the Customers table before you begin to see repeat customers.

FirstPurchaseDate	LastPurchaseDate	Customer Type
2/23/2012	11/24/2015	Repeat Customer
2/23/2012	10/14/2015	Repeat Customer
2/23/2012	4/3/2015	Repeat Customer
2/23/2012	2/23/2012	One-time Customer
2/23/2012	2/23/2012	One-time Customer
2/24/2012	2/24/2012	One-time Customer

11. Now, that you have used the **FirstPurchaseDate** column and the **LastPurchaseDate** column to calculate the value of the **Customer Type** column, you can delete them because they are no longer needed.
 - a) Select the **FirstPurchaseDate** column by clicking its column header.
 - b) Hold down the **SHIFT** key and click the column header for **LastPurchaseDate** so that both columns are selected.
 - c) Right click the one of the selected columns and click the **Remove Columns**.

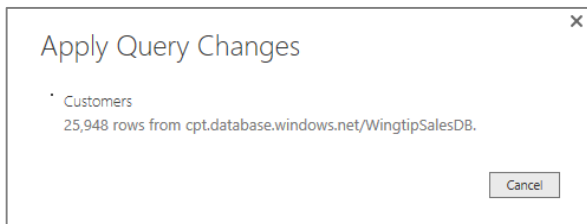
BirthDate	FirstPurchaseDate	LastPurchaseDate	Customer Type
2/16/1982	2/23/2012	11/24/	
6/2/1966	2/23/2012	10/14/	
9/3/1973	2/23/2012	4/3/	
7/19/1963	2/23/2012	2/23/	

- d) You should be able to confirm that the **FirstPurchaseDate** column and the **LastPurchaseDate** columns have been removed from the query results. However, the **Customer Type** column is still there.

Gender	BirthDate	Customer Type
Female	4/11/1966	One-time Customer
Female	6/6/1976	One-time Customer
Female	9/8/1952	One-time Customer
Female	9/12/1939	One-time Customer
Male	9/15/1965	One-time Customer
Male	7/14/1953	One-time Customer
Female	2/3/1992	One-time Customer

You are now done working with the **Customers** query.

12. Click the **Close and Apply** button on the **Home** tab of the ribbon to execute the updated **Customers** query. Power BI Desktop will display the **Apply Query Changes** dialog while importing the data from the SQL Azure database and transforming it as it loads the customer data into the data model.



13. After the query changes have been applied, you should be able to see the results of your changes in the **Customers** table that has been loaded into the project's data model.

WingtipSalesAnalytics - Power BI Desktop

FileHomeModeling

Paste

Cut

Copy

Format Painter

Get Data

Recent Sources

Enter Data

Edit Queries

Refresh

New Page

New Visual

Text Box

Image

Shapes

Page View

Manage Relationships

New Measure

Publish

Clipboard

External Data

Insert

View

Relationships

Calculations

Share

Customerid

City

State

Zipcode

Gender

BirthDate

Customer

Customer Type

760

San Jose

CA

95133

Female

Saturday, March 16, 1968

Lucile Blake

One-time Cu

881

San Jose

CA

95133

Female

Sunday, July 19, 1942

Rochelle Owen

One-time Cu

940

San Jose

CA

95133

Female

Sunday, March 7, 1943

Corinne Finch

One-time Cu

1119

San Jose

CA

95133

Female

Monday, September 3, 1990

Twila Massey

One-time Cu

1548

San Jose

CA

95133

Female

Thursday, July 14, 1955

Kellie Yang

One-time Cu

2195

San Jose

CA

95133

Female

Sunday, March 25, 1951

Megan Martin

One-time Cu

2252

San Jose

CA

95133

Female

Wednesday, April 3, 1946

Cynthia Blake

One-time Cu

2341

San Jose

CA

95133

Female

Monday, May 2, 1960

Karyn Hodges

One-time Cu

Fields

Customers

InvoiceDetails

Invoices

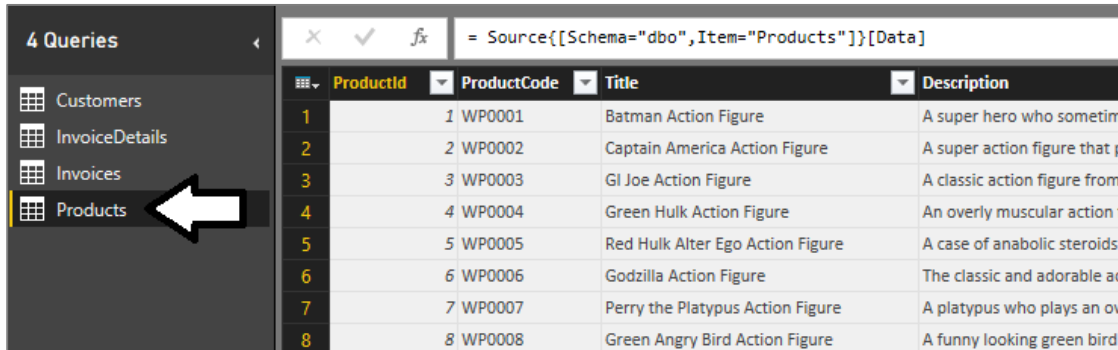
Products

14. Save the work you have done to **WingtipSalesAnalytics.pbix** by clicking the Save button in the upper left corner of the Power BI Desktop window.

Exercise 3: Using Power Query to Transform and Reshape Product Data

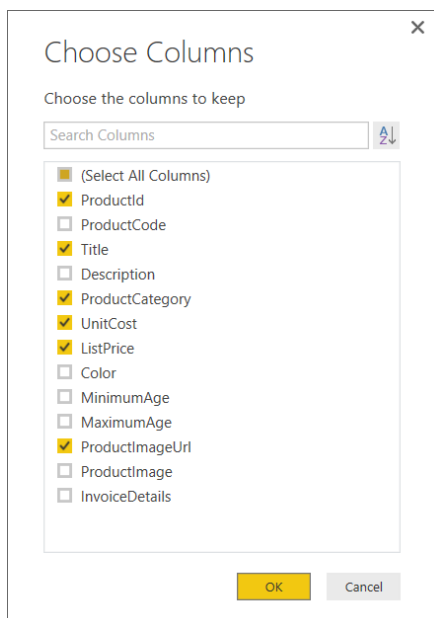
In the following exercise, you will use the **Query Editor** window to modify the **Products** query to perform transforms on product data as it is being loaded into the data model.

1. Make sure you have the **WingtipSalesAnalytics.pbix** project open that you started in the previous exercise.
2. Click on the **Edit Queries** button in the ribbon to display the **Query Editor** window.
3. Make sure the **Products** query is selected in the **Queries** list on the left-hand side of the **Query Editor** window.



	ProductId	ProductCode	Title	Description
1	1	WP0001	Batman Action Figure	A super hero who sometime
2	2	WP0002	Captain America Action Figure	A super action figure that p
3	3	WP0003	GI Joe Action Figure	A classic action figure from
4	4	WP0004	Green Hulk Action Figure	An overly muscular action f
5	5	WP0005	Red Hulk Alter Ego Action Figure	A case of anabolic steroids
6	6	WP0006	Godzilla Action Figure	The classic and adorable ad
7	7	WP0007	Perry the Platypus Action Figure	A platypus who plays an ov
8	8	WP0008	Green Angry Bird Action Figure	A funny looking green bird

4. Remove the columns that are not required in the **Products** query results.
 - a) Click the **Choose Columns** button in the ribbon to display the **Choose Columns** dialog.
 - b) In the **Choose Columns** dialog, begin by clicking on the **(Select all Columns)** checkbox at the top to unselect all columns. Next, select the checkboxes for **ProductId**, **Title**, **ProductCategory**, **UnitCost**, **ListPrice** and **ProductImageUrl** as shown in the following screenshot. Once you have these columns selected, click the **OK** button to close the dialog and to modify the underlying query.



5. Rename the **Title** column to **Product**.
 - a) Right-click on the **Title** column and click **Rename**.
 - b) Update the column name to **Product**.

1 ² ProductId	A ^B _C Product
1	Batman Action Figure
2	Captain America Action Figure
3	GI Joe Action Figure
4	Green Hulk Action Figure
5	Red Hulk Alter Ego Action Figure
6	Godzilla Action Figure
7	Perry the Platypus Action Figure
8	Green Angry Bird Action Figure

6. Split the **ProductCategory** column up into two separate columns named **Category** and **Subcategory**.
 - a) Right-click the **ProductCategory** column and then click the **Split Column > By Delimiter** command.

A ^B _C Product	A ^B _C ProductCategory	1.2 ListPrice	A ^B _C ProductImage
Batman Action Figure	Action Figures > Tough Guys	14.95	http://classres...
Captain America Action Figure	Action Figures > Tough Guys	12.95	http://classres...
GI Joe Action Figure	Action Figures > Tough Guys	14.95	http://classres...
Green Hulk Action Figure	Action Figures > Tough Guys	9.95	http://classres...
Red Hulk Alter Ego Action Figure	Action Figures > Tough Guys	9.95	http://classres...
Godzilla Action Figure	Action Figures > Tough Guys	19.95	http://classres...
Perry the Platypus Action Figure	Action Figures > Cute and Hu	21.95	http://classres...
Green Angry Bird Action Figure	Action Figures > Cute and Hu	4.95	http://classres...
Red Angry Bird Action Figure	Action Figures > Cute and Hu	14.95	http://classres...
Phineas and Ferb Action Figure Set	Action Figures > Cute and Hu	19.95	http://classres...
Black Power Ranger Action Figure	Action Figures > Cute and Hu	7.5	http://classres...
Woody Action Figure	Action Figures > Cute and Hu		

- b) In the **Split Column By Column** dialog, drop down the **Select or enter delimiter** combo box and select **--Custom--**.
- c) In the textbox enter a three-character text value which includes a space follow by the **>** character followed by another space.
- d) When the **Split Column By Column** dialog matches the following screenshot, click the **OK** button.

Split Column by Delimiter

Specify the delimiter used to split the text column.

Select or enter delimiter

--Custom--

>

Split

☐ At the left-most delimiter
 ☐ At the right-most delimiter
 ☒ At each occurrence of the delimiter

Advanced options

OK

Cancel

- e) You should be able to confirm that Power BI Desktop has split the **ProductCategory** column into two separate columns named **ProductCategory.1** and **ProductCategory.2**.

A ^B _C Product	A ^B _C ProductCategory.1	A ^B _C ProductCategory.2
Batman Action Figure	Action Figures	Tough Guys
Captain America Action Figure	Action Figures	Tough Guys
GI Joe Action Figure	Action Figures	Tough Guys
Green Hulk Action Figure	Action Figures	Tough Guys
Red Hulk Alter Ego Action Figure	Action Figures	Tough Guys
Godzilla Action Figure	Action Figures	Tough Guys

- f) Rename the **ProductCategory.1** column to **Category** and rename **ProductCategory.2** to **Subcategory**.

A ^B _C Product	A ^B _C Category	A ^B _C Subcategory
Batman Action Figure	Action Figures	Tough Guys
Captain America Action Figure	Action Figures	Tough Guys
GI Joe Action Figure	Action Figures	Tough Guys
Green Hulk Action Figure	Action Figures	Tough Guys
Red Hulk Alter Ego Action Figure	Action Figures	Tough Guys
Godzilla Action Figure	Action Figures	Tough Guys
Perry the Platypus Action Figure	Action Figures	Cute and Huggable

When you have query columns based on numeric currency values, it is best to change their column type to **Fixed Decimal Number**.

7. Modify the column type of the **UnitCost** column and the **ListPrice** column to the **Fixed Decimal Number** type.
- a) Use the dropdown column type menu to set the type of the **UnitCost** to **Fixed Decimal Number**.

1.2 UnitCost	1.2 ListPrice
1.2 Decimal Number	14.95
\$ Fixed Decimal Number	12.95
1 ² ₃ Whole Number	14.95
Date/Time	9.95
Date	9.95
Time	19.95

- b) Use the dropdown column type menu to set the type of the **ListPrice** to **Fixed Decimal Number**.

\$ UnitCost	\$ ListPrice
6.85	14.95
7.05	12.95
6.1	14.95
2.85	9.95
2.85	9.95

You are now done working with the **Products** query.

8. Click the **Close and Apply** button on the **Home** tab of the ribbon to execute the updated **Products** query. Power BI Desktop will display the **Apply Query Changes** dialog while importing the data and transforming it to load it into the data model.
9. After the query changes have been applied, you should be able to see the results of your changes in the **Products** table that has been loaded into the project's data model.

ProductId	Product	UnitCost	ListPrice	ProductImageUrl	Category	Subcategory
1	Batman Action Figure	6.85	14.95	http://classresources.blob.core.windows.net/images/WP0001.jpg	Action Figures	Tough Guys
2	Captain America Action Figure	7.05	12.95	http://classresources.blob.core.windows.net/images/WP0002.jpg	Action Figures	Tough Guys
3	GI Joe Action Figure	6.1	14.95	http://classresources.blob.core.windows.net/images/WP0003.jpg	Action Figures	Tough Guys
4	Green Hulk Action Figure	2.85	9.95	http://classresources.blob.core.windows.net/images/WP0004.jpg	Action Figures	Tough Guys
5	Red Hulk Alter Ego Action Figure	2.85	9.95	http://classresources.blob.core.windows.net/images/WP0005.jpg	Action Figures	Tough Guys
6	Godzilla Action Figure	14.25	19.95	http://classresources.blob.core.windows.net/images/WP0006.jpg	Action Figures	Tough Guys
7	Perry the Platypus Action Figure	12	21.95	http://classresources.blob.core.windows.net/images/WP0007.jpg	Action Figures	Cute and Huggable

You might have noticed that the columns that are displayed in the **Products** table do not appear in the same order as they appear in the **Products** query. Power BI Desktop currently provides no ability to control and change the order of columns or fields as they are defined in a table. Microsoft is expected to address this limitation soon in a future update to the Power BI Designer.

10. Save your work by clicking the Save button in the upper left corner of the Power BI Desktop window.

Exercise 4: Using Power Query to Transform and Reshape Product Data

In the following exercise, you will use the Query Editor to modify the **InvoiceDetails** query to transform data related to sales data before it is loaded into the data model.

1. Make sure you have the **WingtipSalesAnalytics.pbix** project open that you started in the previous exercise.
2. Click on the **Edit Queries** button in the ribbon to display the Query Editor window.
3. Make sure the **InvoiceDetails** query is selected in the **Queries** list on the left-hand side of the **Query Editor** window.

Id	Quantity	SalesAmount	InvoiceId	ProductId	Invoices	Products
1	1	2.95	1	16	Value	Value
2	6	14.94	2	15	Value	Value
3	1	19.95	3	6	Value	Value
4	5	249.75	3	17	Value	Value
5	1	2.95	4	16	Value	Value
6	1	12.95	4	19	Value	Value
7	9	22.41	5	15	Value	Value
8	7	349.65	5	17	Value	Value
9	7	90.65	6	19	Value	Value

One important point of flexibility in the import process is that you can change the name of a query, and therefore the name of the resulting table to make the data model more intuitive and easier to understand. The **InvoiceDetails** query is returning data that will be used to calculate sales results at the most granular level. Therefore, the data model will be easier to understand if you change the name the name of the **InvoiceDetails** query to **Sales**. This will result in renaming the resulting table to the **Sales** table.

4. Update the name of the **InvoiceDetails** query to **Sales** by replacing the text in the **Name** textbox in the **Query Settings** pane.

Id	Quantity	SalesAmount	InvoiceId	ProductId	Invoices	Products
1	1	2.95	1	16	Value	Value
2	6	14.94	2	15	Value	Value
3	1	19.95	3	6	Value	Value
4	5	249.75	3	17	Value	Value
5	1	2.95	4	16	Value	Value
6	1	12.95	4	19	Value	Value
7	9	22.41	5	15	Value	Value
8	7	349.65	5	17	Value	Value

5. Using the mouse, drag and drop the column header for the **InvoiceId** column to move it to the immediate right of the **Id** column.

	1 ² ₃ Id	1 ² ₃ InvoiceId	1 ² ₃ Quantity	1.2 SalesAmount
1	1	1	1	2.95
2	2	2	6	14.94
3	3	3	1	19.95
4	4	3	5	249.75
5	5	4	1	2.95
6	6	4	1	12.95

6. Using the mouse, drag and drop the column header for the **ProductId** column to move it to the right of the **InvoiceId** column.

	1 ² ₃ Id	1 ² ₃ InvoiceId	1 ² ₃ ProductId	1 ² ₃ Quantity
1	1	1	16	1
2	2	2	15	6
3	3	3	6	1
4	4	3	17	5
5	5	4	16	1
6	6	4	19	1

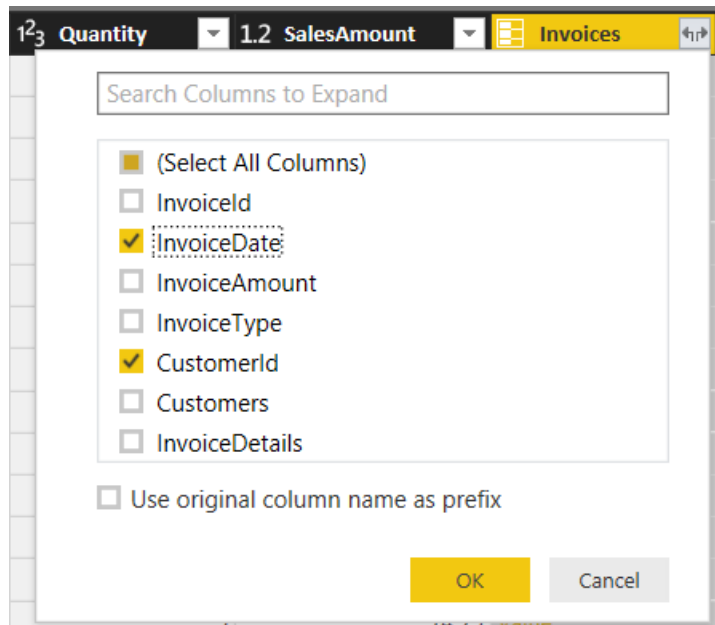
7. Modify the column type of the **SalesAmount** column to the **Fixed Decimal Number** type.

1 ² ₃ Quantity	1.2 SalesAmount	Invoices
1	1.2 Decimal Number	
6	\$ Fixed Decimal Number	
1	1 ² ₃ Whole Number	
5	Date/Time	
1	Date	
1	Time	
9	Date/Time/Timezone	

8. Expand the **Invoices** column to add the **InvoiceDate** column and the **CustomerId** column to the **Sales** query.
- a) Click the **Expand** button inside the column header of the **Invoices** column to display the **Columns to Expand** dialog.

1 ² ₃ Quantity	1.2 SalesAmount	Invoices	ts
1	2.95	Value	Value
6	14.94	Value	Value
1	19.95	Value	Value

- b) In the **Columns to Expand** dialog, begin by clicking on the **(Select all Columns)** checkbox at the top to unselect all columns. Next, select the checkboxes for the **InvoiceDate** column and the **CustomerId** column. Also make sure to uncheck the checkbox with the caption **Use original column name as prefix**. Once the **Columns to Expand** dialog looks like the one shown in the following screenshot, click the **OK** button to close the dialog and to modify the underlying query.



c) You should see that the **InvoiceDate** column and the **CustomerId** column have now been added to the **Sales** query results.

1.2 SalesAmount	InvoiceDate	1.2 CustomerId	Products
2.95	1/28/2012 12:00:00 AM	1	Value
14.94	1/28/2012 12:00:00 AM	2	Value
19.95	1/28/2012 12:00:00 AM	3	Value
249.75	1/28/2012 12:00:00 AM	3	Value
2.95	1/28/2012 12:00:00 AM	4	Value
12.95	1/28/2012 12:00:00 AM	4	Value

9. Use the mouse to drag and drop the **CustomerId** column to move it to the right of the **InvoiceId** column.

1.2 Id	1.2 InvoiceId	1.2 CustomerId	1.2 ProductId
1	1	1	16
2	2	2	15
3	3	3	6
4	4	3	17
5	5	4	16

10. Use the mouse to drag and drop the **InvoiceDate** column to move it to the right of the **ProductId** column.

1.2 Id	1.2 InvoiceId	1.2 CustomerId	1.2 ProductId	InvoiceDate	1.2 Quantity
1	1	1	16	1/28/2012 12:00:00 AM	1
2	2	2	15	1/28/2012 12:00:00 AM	6
3	3	3	6	1/28/2012 12:00:00 AM	1
4	4	3	17	1/28/2012 12:00:00 AM	5
5	5	4	16	1/28/2012 12:00:00 AM	1
6	6	4	19	1/28/2012 12:00:00 AM	1

11. Change the column type of the **InvoiceDate** to the **Date** type.

	1 ² ₃ Id	1 ² ₃ InvoiceId	1 ² ₃ CustomerId	1 ² ₃ ProductId	InvoiceDate	1 ² ₃ Quantity
1	1	1	1	16	1.2 Decimal Number	
2	2	2	2	15	\$ Fixed Decimal Number	
3	3	3	3	6	1 ² ₃ Whole Number	
4	4	3	3	17	Date/Time	
5	5	4	4	16	Date ←	
6	6	4	4	19	Time	
7	7	5	5	15	Date/Time/Timezone	

12. Change the name of the **InvoiceDate** column to **PurchaseDate**.

	1 ² ₃ Id	1 ² ₃ InvoiceId	1 ² ₃ CustomerId	1 ² ₃ ProductId	PurchaseDate	1 ² ₃ Quantity
1	1	1	1	16	1/28/2012	1
2	2	2	2	15	1/28/2012	6
3	3	3	3	6	1/28/2012	1
4	4	3	3	17	1/28/2012	5
5	5	4	4	16	1/28/2012	1
6	6	4	4	19	1/28/2012	1

13. Expand the **Products** column to add the **UnitCost** column to the **Sales** query.

a) Click the Expand button inside the column header of the **Products** column to display the **Columns to Expand** dialog.

1.2 SalesAmount	Products
2.95	Value
14.94	Value
19.95	Value

b) In the **Columns to Expand** dialog, begin by clicking on the **(Select all Columns)** checkbox at the top to unselect all columns. Next, select the checkbox for the **UnitCost** column. Also make sure to uncheck the checkbox with the caption **Use original column name as prefix**. Once the **Columns to Expand** dialog looks like the one shown in the following screenshot, click the **OK** button to close the dialog and to modify the underlying query.

Quantity
1.2 SalesAmount
Products

Search Columns to Expand

☒ (Select All Columns)
☐ ProductId
☐ ProductCode
☐ Title
☐ Description
☐ ProductCategory
☒ UnitCost
☐ ListPrice
☐ Color
☐ MinimumAge
☐ MaximumAge
☐ ProductImageUrl
☐ ProductImage
☐ InvoiceDetails

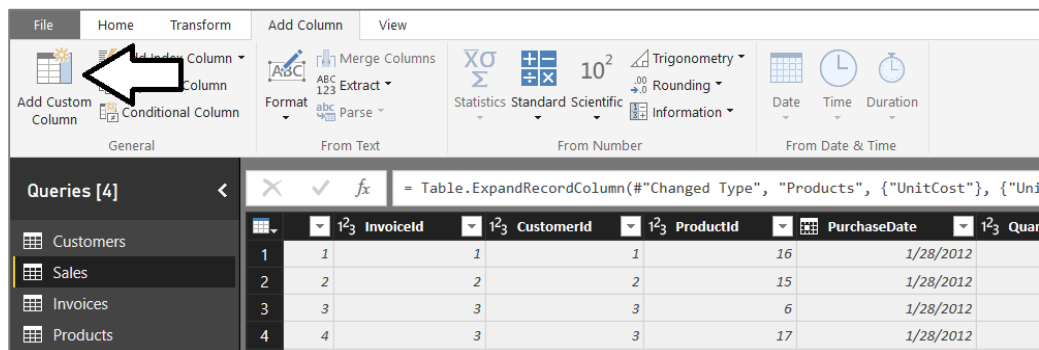
☐ Use original column name as prefix

OK Cancel

c) You should see that the **UnitCost** column has now been added to the **Sales** query results.

1 ² ₃ Quantity	1.2 SalesAmount	1.2 UnitCost
1	2.95	0.85
6	14.94	1.2
1	19.95	14.25
5	249.75	12.1
1	2.95	0.85
1	12.95	7.25

14. Add a new custom column named **ProductCost** to calculate the product of the **Quantity** field multiplied by the **UnitCost** field.
- Activate the **Add Column** tab in the ribbon.
 - Click the **Add Custom Column** button in the ribbon to display the **Add Custom Column** dialog.



- In the **Add Custom Column** dialog, add a value of **ProductCost** in the **New column name** textbox.
- In the **Custom column formula** textbox, enter the following formula.

[Quantity] * [UnitCost]

- When the **Add Custom Column** dialog appears as the following screenshot, click the **OK** button to add the new column.

Add Custom Column

New column name

Custom column formula:

[Learn about Power BI Desktop formulas](#)

✓ No syntax errors have been detected.

Available columns:

- InvoiceId
- CustomerId
- ProductId
- PurchaseDate
- Quantity
- SalesAmount
- UnitCost

<< Insert

OK

Cancel

- You should be able to verify that the new **ProductCost** column has a value calculated by multiplying the value of the **Quantity** column together with the value of the **UnitCost** column.

1 ² ₃ Quantity	1.2 SalesAmount	1.2 UnitCost	ABC 123 ProductCost
1	2.95	0.85	0.85
6	14.94	1.2	7.2
1	19.95	14.25	14.25
5	249.75	12.1	60.5

Once the **UnitCost** column has been used to calculate the value for the **ProductCost** value, this column is no longer needed and can be removed from the results of the **Sales** query.

- Remove the **UnitCost** column by right-clicking its column header and selecting the **Remove** command.

1.2 SalesAmount	1.2 UnitCost	ABC 123 ProductCost
2.95		
14.94		
19.95		
249.75		

- Modify the column type of the **SalesAmount** column and the **ProductCost** column to the **Fixed Decimal Number** type. Once you have done this, you should be able to see dollar signs to the left of the column names.

1 ² ₃ Quantity	\$ SalesAmount	\$ ProductCost
1	2.95	0.85
6	14.94	7.2
1	19.95	14.25
5	249.75	60.5
1	2.95	0.85

You are now done working with the **Sales** query.

- Click the **Close and Apply** button on the **Home** tab of the ribbon to execute the updated **Sales** query. Power BI Desktop will display the **Apply Query Changes** dialog while importing the data and transforming it to load it into the data model.
- After the query changes have been applied, you should be able to see the results of your changes in the **Sales** table that has been loaded into the project's data model.

Id	Quantity	SalesAmount	InvoiceId	ProductId	CustomerId	PurchaseDate	ProductCost
95	9	179.55	46	6	46	Saturday, February 4, 2012	\$128.25
96	9	179.55	47	6	47	Saturday, February 4, 2012	\$128.25
307	9	179.55	155	6	142	Thursday, February 23, 2012	\$128.25
313	9	179.55	157	6	114	Thursday, February 23, 2012	\$128.25
357	9	179.55	180	6	116	Saturday, February 25, 2012	\$128.25
601	9	179.55	296	6	240	Saturday, March 10, 2012	\$128.25
617	9	179.55	306	6	130	Saturday, March 10, 2012	\$128.25
761	9	179.55	378	6	297	Monday, March 19, 2012	\$128.25
861	9	179.55	427	6	325	Saturday, March 24, 2012	\$128.25
863	9	179.55	428	6	326	Saturday, March 24, 2012	\$128.25
864	9	179.55	429	6	327	Saturday, March 24, 2012	\$128.25

Unfortunately, the current version of Power BI Desktop does not retain the order of the columns that you configured in the **Sales** query.

19. Save your work by clicking the Save button in the upper left corner of the Power BI Desktop window.

Exercise 5: Using Power Query to Transform and Reshape Purchase Data

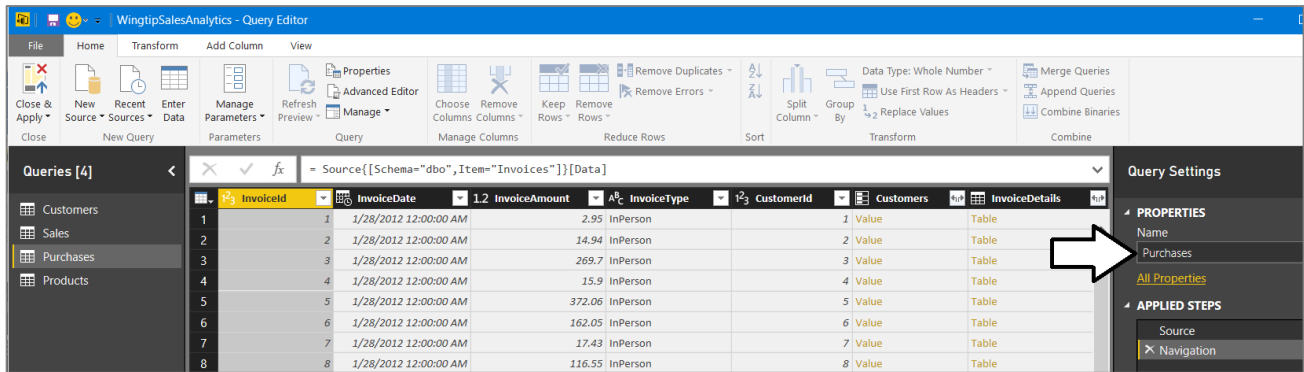
In the following exercise, you will use the **Query Editor** window to modify the **Invoices** query to transform invoice data as it is being loaded into the data model.

1. Make sure you have the **WingtipSalesAnalytics.pbix** project open that you started in the previous exercise.
2. Click on the **Edit Queries** button in the ribbon to display the **Query Editor** window.
3. Make sure the **Invoices** query is selected in the **Queries** list on the left-hand side of the Query Editor window.

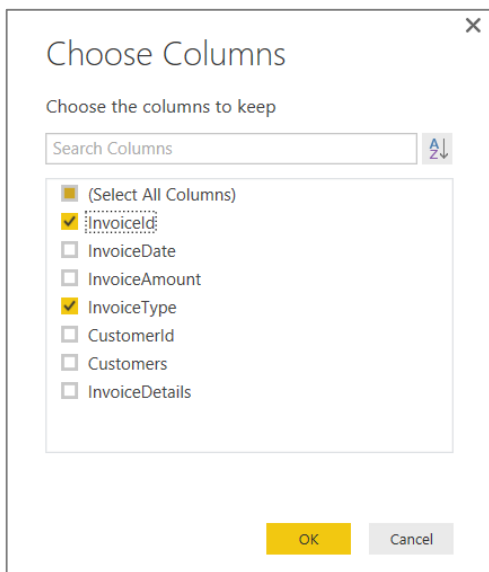
InvoiceId	InvoiceDate	InvoiceAmount	InvoiceType	CustomerId	InvoiceDetails
1	1/28/2012 12:00:00 AM	2.95	InPerson	1	Value Table
2	1/28/2012 12:00:00 AM	14.94	InPerson	2	Value Table
3	1/28/2012 12:00:00 AM	269.7	InPerson	3	Value Table
4	1/28/2012 12:00:00 AM	15.9	InPerson	4	Value Table
5	1/28/2012 12:00:00 AM	372.06	InPerson	5	Value Table
6	1/28/2012 12:00:00 AM	162.05	InPerson	6	Value Table
7	1/28/2012 12:00:00 AM	17.43	InPerson	7	Value Table
8	1/28/2012 12:00:00 AM	116.55	InPerson	8	Value Table

In the last exercise you changed the name of the **InvoiceDetails** to **Sales** to make the data model easier to understand. In this exercise you will change the name of the **Invoices** query to **Purchases** for the same reason.

4. Update the name of the **Invoices** query to **Purchases** by replacing the text in the **Name** textbox in the **Query Settings** pane.



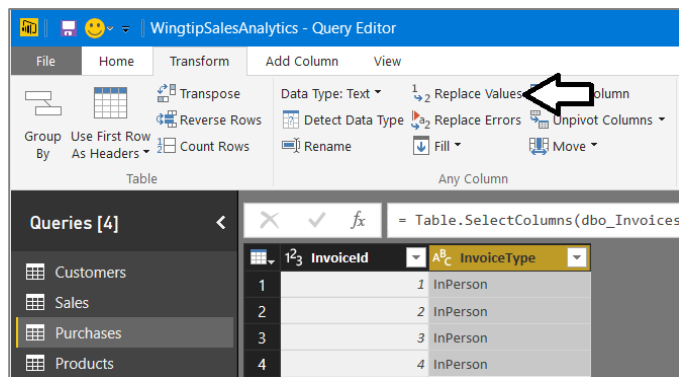
5. Make sure the **Purchases** query is selected in the **Queries** list on the left-hand side of the Query Editor window. Click the **Choose Columns** button in the ribbon to display the **Choose Columns** dialog.
6. In the **Choose Columns** dialog, begin by clicking on the **(Select all Columns)** checkbox at the top to unselect all columns. Next, select the checkboxes for **InvoiceId** and **InvoiceType** as shown in the following screenshot. Once you have these columns selected, click the **OK** button to close the **Choose Columns** dialog and to modify the underlying query.



7. You should be able to see that the Query Editor window now only shows the columns that you selected.

	1 ² InvoiceId	A ^B InvoiceType
1		1 InPerson
2		2 InPerson
3		3 InPerson
4		4 InPerson
5		5 InPerson
6		6 InPerson
7		7 InPerson
8		8 InPerson

8. Modify the query so that the **InvoiceType** column returns values that are more human readable.
 - a) Make sure the **Transform** tab is the active tab in the ribbon.
 - b) Select the **InvoiceType** column by clicking its column header.
 - c) Click the **Replace Values** button in the ribbon to display the **Replace Values** dialog.



- d) In the **Replace Value** dialog, enter a value of **InPerson** in the **Value to Find** textbox and enter a value of **Store Purchase** in the **Replace With** textbox. Click to **OK** button add your changes to the underlying query.

Replace Values

Replace one value with another in the selected columns.

Value To Find

InPerson

Replace With

Store Purchase

Advanced options

OK

Cancel

- e) Make sure the **InvoiceType** column is still selected.
- f) Click the **Replace Values** button in the ribbon to display the **Replace Values** dialog.
- g) In the **Replace Value** dialog, enter a value of **MailOrder** in the **Value to Find** textbox and enter a value of **Mail Order Purchase** in the **Replace With** textbox. Click to **OK** button add your changes to the underlying query.

Replace Values

Replace one value with another in the selected columns.

Value To Find

MailOrder

Replace With

Mail Order Purchase

Advanced options

OK

Cancel

- h) Make sure the **InvoiceType** column is still selected.
- i) Click the **Replace Values** button in the ribbon to display the **Replace Values** dialog.
- j) In the **Replace Value** dialog, enter a value of **Online** in the **Value to Find** textbox and enter a value of **Online Purchase** in the **Replace With** textbox. Click to **OK** button add your changes to the underlying query.

Replace Values

Replace one value with another in the selected columns.

Value To Find

Online

Replace With

Online Purchase

Advanced options

OK

Cancel

- k) If you scroll down and look at all the rows within the **Purchases** table, you should be able to see that each row has a **InvoiceType** column value of either **Store Purchase**, **Mail Order Purchase** or **Online Purchase**.

9. Change the name of the **InvoiceType** column to **PurchaseType**.

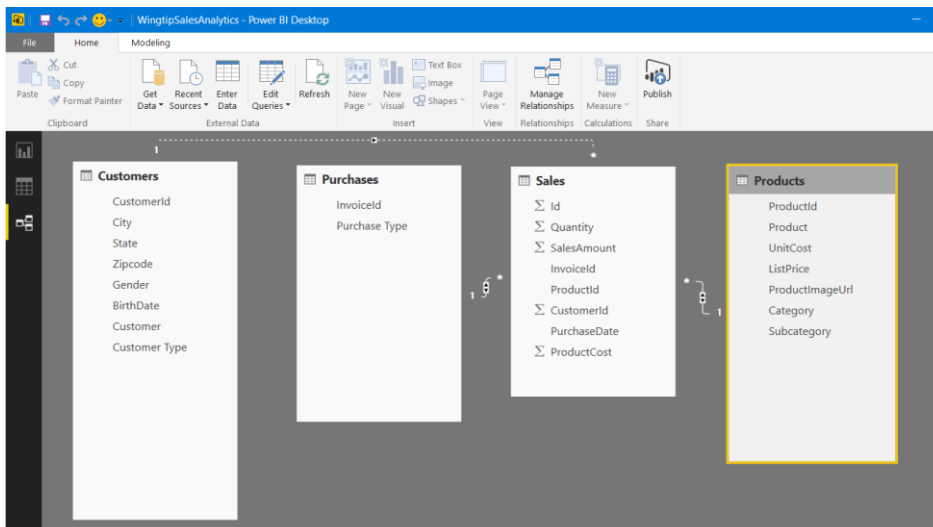
	1 ² InvoiceId	AB ^C Purchase Type
1		1 Store Purchase
2		2 Store Purchase
3		3 Store Purchase
4		4 Store Purchase
5		5 Store Purchase
6		6 Store Purchase
7		7 Store Purchase
8		8 Store Purchase

You are now done working with the **Purchase** query.

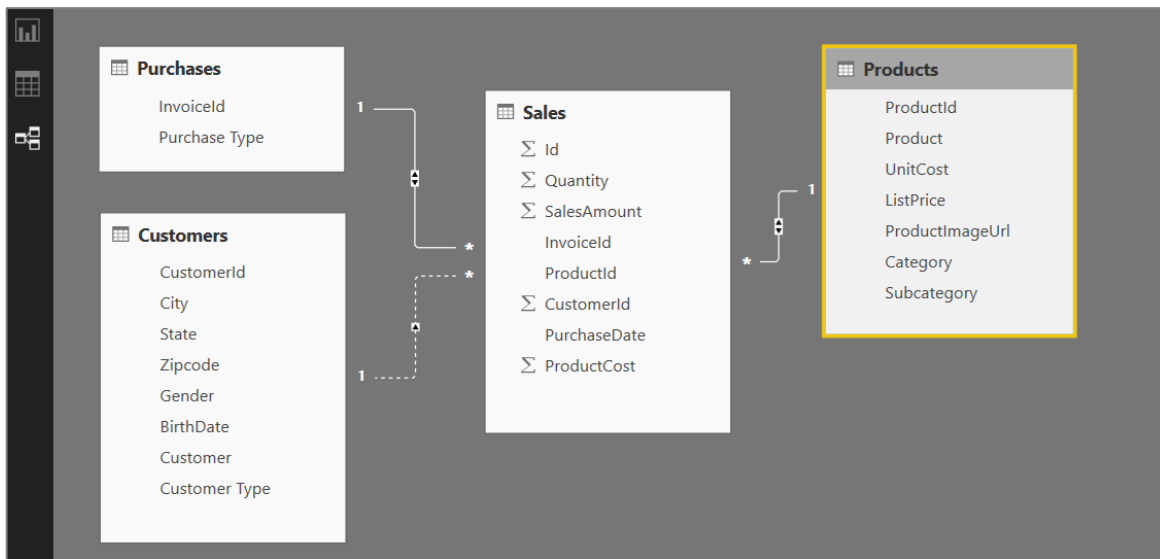
10. Click the **Close and Apply** button on the **Home** tab of the ribbon to execute the updated **Purchases** query. Power BI Desktop will display the **Apply Query Changes** dialog while importing the data and transforming it to load it into the data model.
11. After the query changes have been applied, you should be able to see the results of your changes in the **Purchase** table that has been loaded into the project's data model.

InvoiceId	Purchase Type
1	Store Purchase
2	Store Purchase
3	Store Purchase
4	Store Purchase
5	Store Purchase
6	Store Purchase

12. Navigate to relationship view to see how your updates to the queries in this project have affected the relationships between the tables. You should be able to see that there is no longer a relationship between the **Customers** table and the **Purchases** table. This is due to the fact that you removed the **CustomerId** column from the **Purchases** table.



13. Use your mouse to move and resize the four tables in the data model to match the following screenshot.

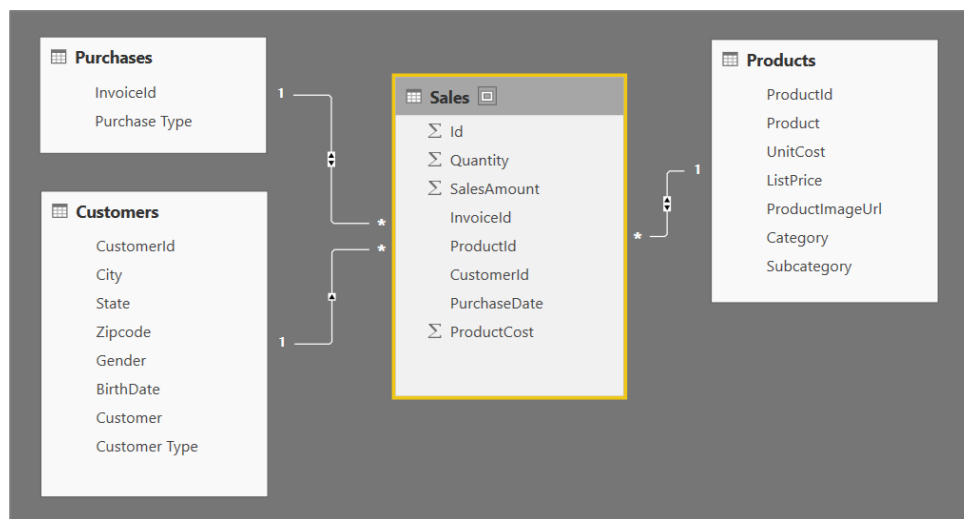


14. Modify the relationship between the **Customers** table and the **Sales** table.

- Notice the relationship between Customers and sales has a dotted line instead of a solid line. This means the relationship has been created but it has not been configured as an active relationship.
- Double-click the dotted line between the **Customers** table and the **Sales** table to edit the relationship.
- In the **Edit Relationship** dialog, click the **Make this relationship active** checkbox so it is checked and then click **OK**.

The screenshot shows the 'Edit Relationship' dialog box. The 'Sales' table is selected, and the 'Customers' table is selected. The dialog displays the columns for both tables. The 'Cardinality' is set to 'Many to One (*:1)' and the 'Cross filter direction' is set to 'Single'. The 'Make this relationship active' checkbox is checked, indicated by a yellow arrow. The 'Assume Referential Integrity' checkbox is unchecked. The 'OK' and 'Cancel' buttons are at the bottom.

- You should see a solid line between the **Customers** table and the **Sales** tables indicating that the relationship between these two tables is now an active relationship.



The tables in the data model have now been refactored into a "star schema" that is commonly used in data modeling for BI projects.

15. Save your work by clicking the Save button in the upper left corner of the Power BI Desktop window.

You are now done with your work modifying queries to transform data as it is being loaded into the data model of your project. Note that you will continue to work on the PBIX file with this project in the next lab and the ones that follow.