DAT225x

Developing an Analysis Services Tabular Model

Lab 02 | Creating the Tabular Project

Estimated time to complete this lab is 60 minutes

Overview

In this lab, you will commence the development of a Tabular Project based on the SQL Server **AdventureWorksDW2016** database. This will involve preparing data, importing tables and configuring relationships.

Note: The four labs in this course are accumulative. You cannot complete this lab if you did not successfully complete **Lab 01**.
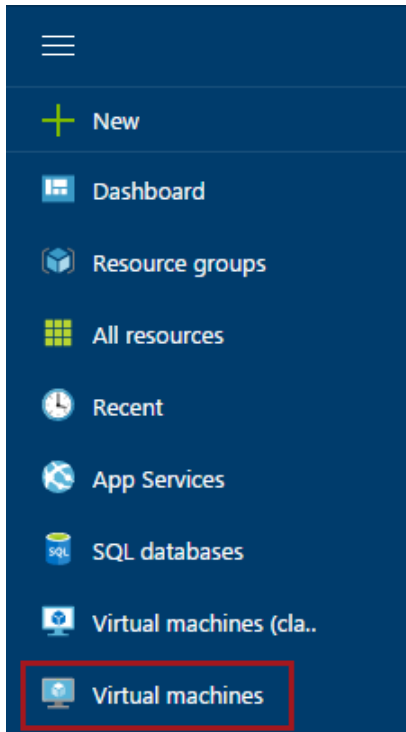
Getting Started

In this exercise, you will get started with the VM created in **Lab 01**.

Getting Started

In this task, you will start the VM, and then connect to it to complete the exercises in this lab.

1. Sign in to the **Azure Portal** by using your subscription.

2. In the left pane, select **Virtual Machines**—do not select **Virtual Machines (Classic)**.
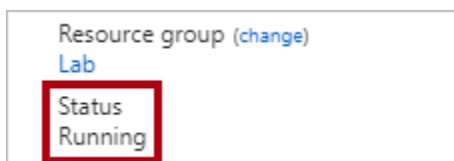

3. In the **Virtual Machines** blade, select the VM you provisioned in **Lab 01**.

4. In the VM blade, click **Start**.

5. Wait for the VM status to update to **Running**.

    It usually takes 1-2 minutes for the VM to start.

6.   To connect to the VM, click **Connect**.

Take care not to use the RDP file downloaded in the previous lab. It is likely that a different IP address has be assigned.


7.   In the **Connect to Virtual Machine** pane (located at the right), click **Dowload RDP File**.


A Remote Desktop File (.rdp) file is downloaded to your computer.

This file can be used to reconnect to the remote desktop session, but note that if you deallocate the VM and later re-start the VM, it will be likely that a different IP address will be assigned.

8.   If prompted by the web browser to open the Remote Desktop File, click **Open**, otherwise, locate the downloaded file, and then double-click it.
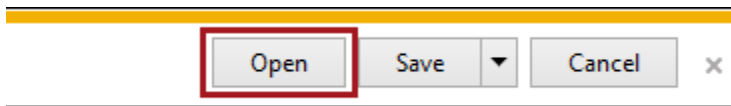
9.   If prompted to connect to the unknown publisher, click **Connect**.

10. If prompted, in the **Windows Security** dialog window, click **Use Another Account**.
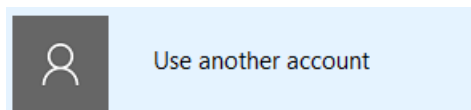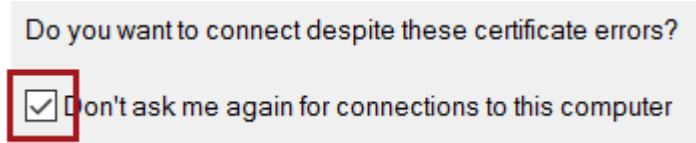

11. Enter the credentials you created for your VM.

12. Check the **Remember My Credentials** checkbox.
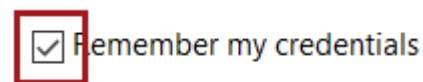
13. Click **OK**.

14. In the **Remote Desktop Connection** dialog window, check the
    Don't Ask Me Again for Connections to This Computer checkbox.

    Do you want to connect despite these certificate errors?

   ☑ Don't ask me again for connections to this computer

15. Click **Yes**.

Exercise 1: Creating the Tabular Project

In this exercise, you will create the Tabular Project.

Configuring the Analysis Services Options

In this task, you will launch SSDT and configure options that will be applied to new Tabular Projects.

1. To launch SSDT, on the task bar, click the **Visual Studio 2017 (SSDT)** shortcut.
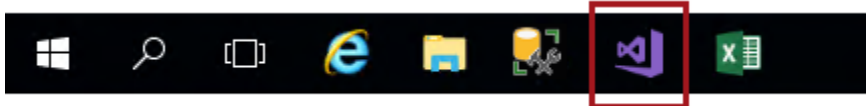
2. To configure the Analysis Services options, on the **Tools** menu, select **Options**.

3. In the **Options** window, expand the **Analysis Services Tabular** group (you will need to scroll down the list to locate this group), and then select the **Deployment** page.
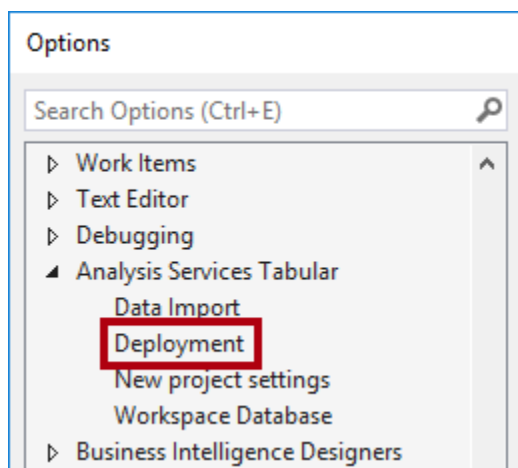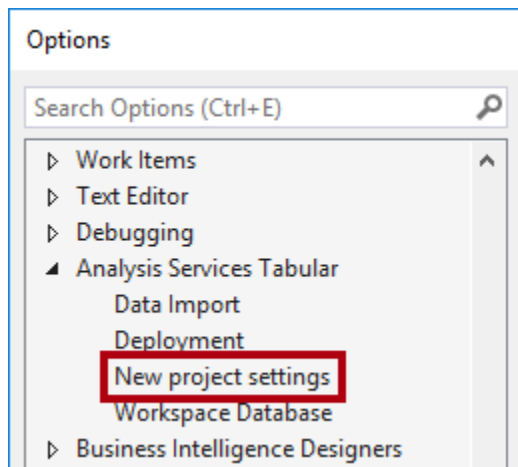

4. In the **Default Deployment Server** dropdown list, verify that **localhost** is entered.

    *This is the Analysis Services instance that you started in **Lab 01**.*
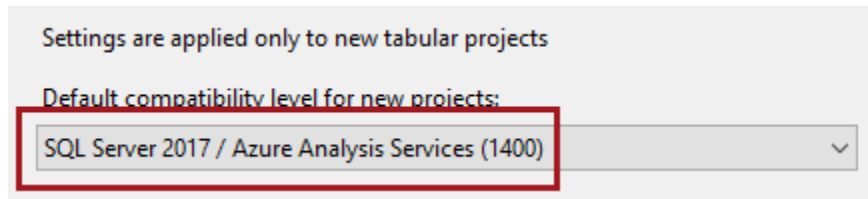

5. Click **Test Connection**.

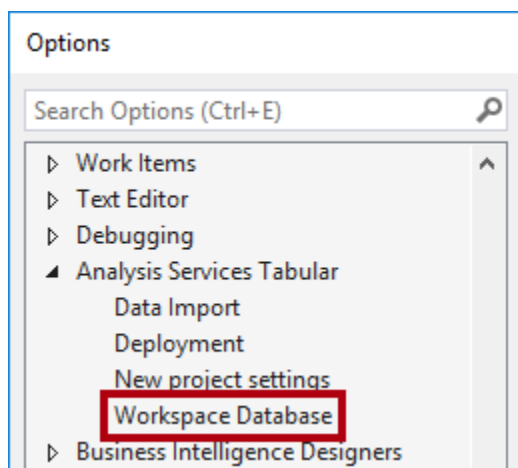6. When the connection test has succeeded, click **OK**.

7. Select the **New Project Settings** page.

8. In the **Default Compatibility Level for New Projects** dropdown list, select **SQL Server 2017 / Azure Analysis Services (1400)**.

9. Select the **Workspace Database** page.

10. In the **Workspace Server** dropdown list, verify that **localhost** is entered.

When you work with the tabular model designer, you are working with a temporary Analysis Services database that automatically loads on a workspace server. It is also possible to use an integrated workspace which uses an internal Analysis Services instance in the background, and so does not require an instance of Analysis Services to be installed.
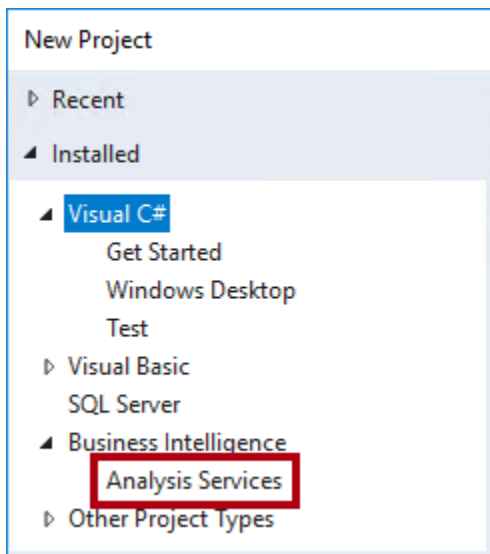
In this lab, you will work with the workspace server.
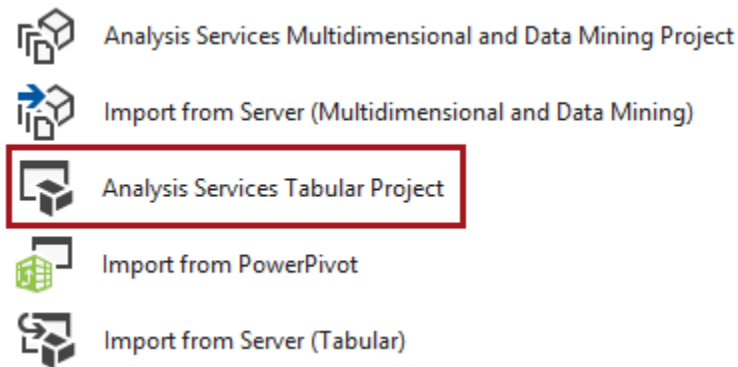
11. Click **OK**.

Creating the Tabular Project

In this task, you will create the Tabular Project.

1. To create a solution, on the **File** menu, select **New | Project**.

2. In the **New Project** window, in the left pane, from inside the **Business Intelligence** group, select the **Analysis Services** template.
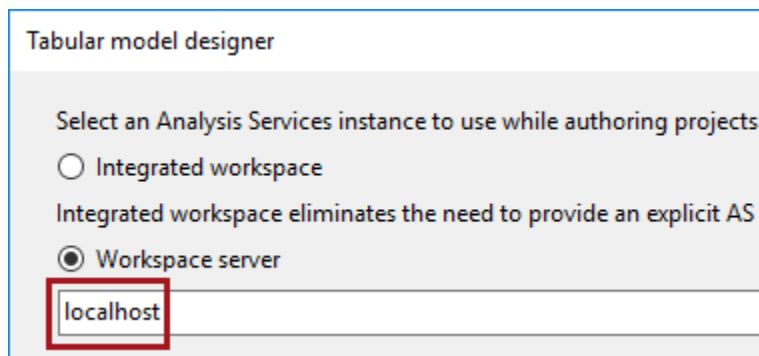
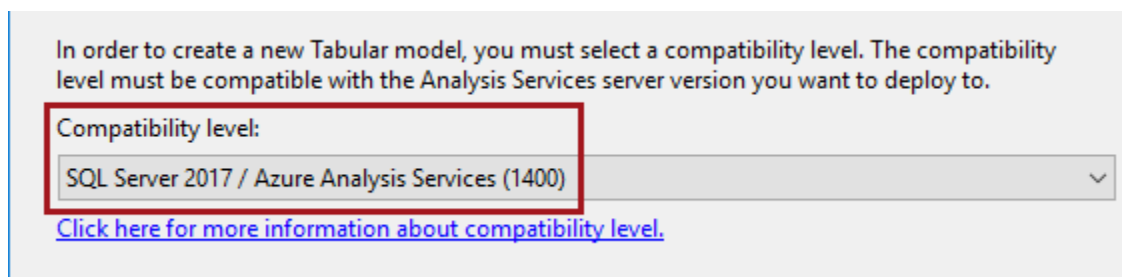3. Select the **Analysis Services Tabular Project** template.

4. In the **Name** box, replace the text with **Reseller Sales**.

5. In the **Solution Name** box, replace the text with **AdventureWorksBI**.
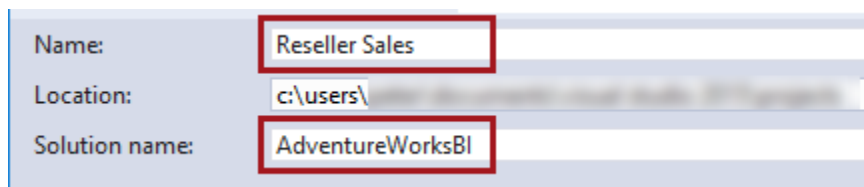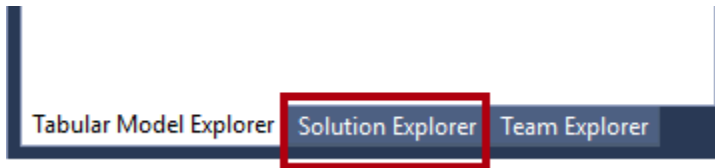

6. Click **OK**.

7. In the **Tabular Model Designer** window, notice that the project will use the default workspace server.
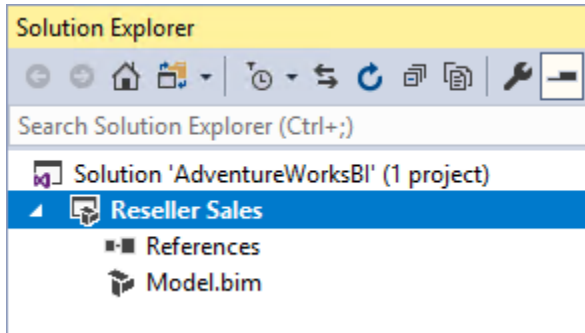

8. Notice that the **Compatibility Level** property is set to **SQL Server 2017**.

9. Click **OK**.

10. In the right pane, select **Solution Explorer**.

11. Notice that the **Reseller Sales** project consists of a single project item named **Model.bim**.

*The **Model.bim** item is the data model that you will commence developing in this lab.*

*Each tabular project consists of a single data model, and no additional data models can be added. When deployed for the first time, the project creates a database on the target Analysis Services instance. You will deploy the project in **Lab 04**.*

12. Notice that the **Model.bim** item was automatically opened upon project creation.
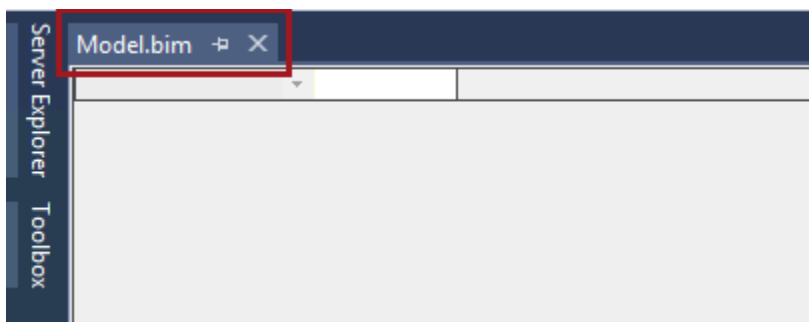
13. To save the project, on the **File** menu, select **Save All**.

It is a good practice to regularly save the solution to protect your development effort in case of an unexpected application crash.

*The **Save All** function is also available on the toolbar.*

Exercise 2: Importing Data

In this exercise, you will import data from two data sources. The first will be the SQL Server **AdventureWorksDW2016** database; the second will be a CSV file.

You will create one query for each table that will be loaded to the model. You will design the queries to only include columns and rows that are required by the model. This design approach will conserve server resources (specifically memory) and help speed up query response times.

Connecting to SQL Server

In this task, you will import several tables from a SQL Server relational database.

1. To manage the tabular model design, in the right pane, select **Tabular Model Explorer**.
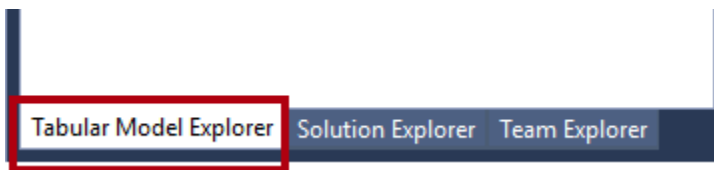

   This pane lets you conveniently navigate through the various metadata objects in a model, including data sources, tables, measures, and relationships.

2. To connect to SQL Server, in **Tabular Model Explorer**, right-click the **Data Sources** folder, and then select **Import from Data Source**.
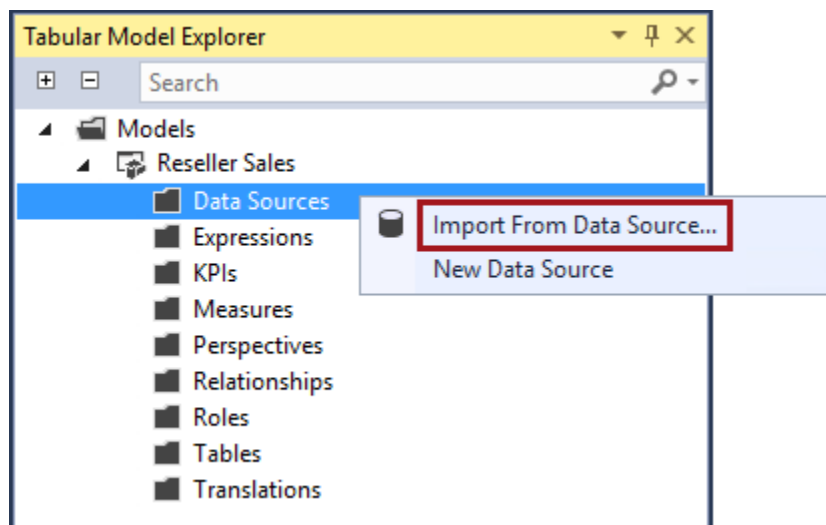
   *The context menu functions available in **Tabular Model Explorer** are also available on the Visual Studio **Model**, **Table** and **Column** menus.*
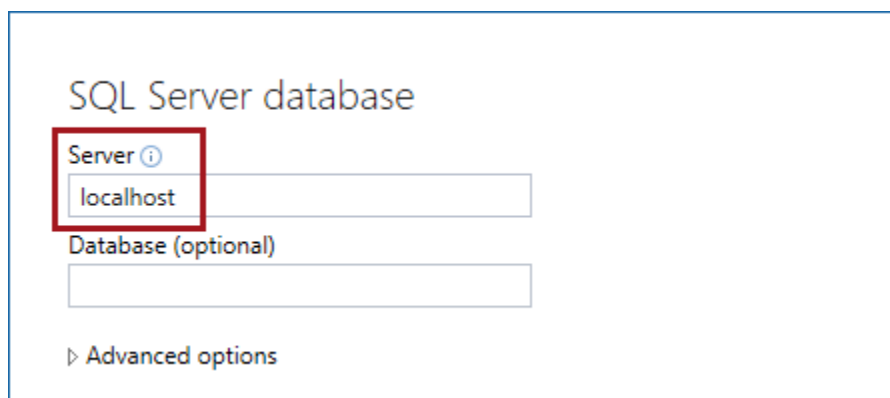
3. In the **Get Data** window, select the **SQL Server Database** source.
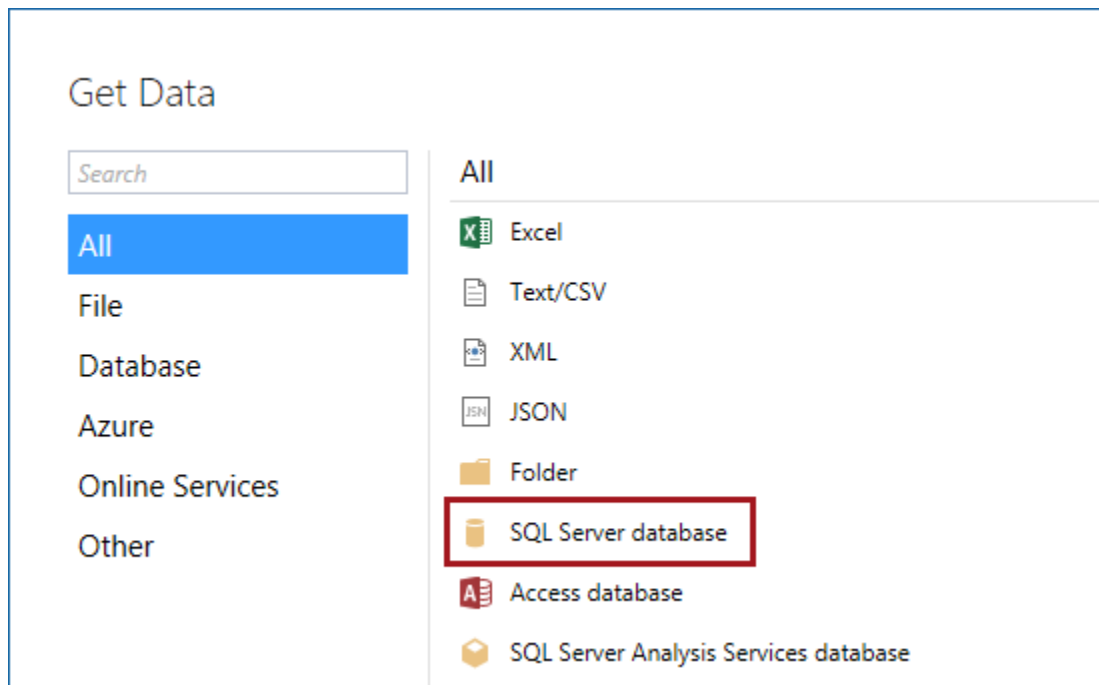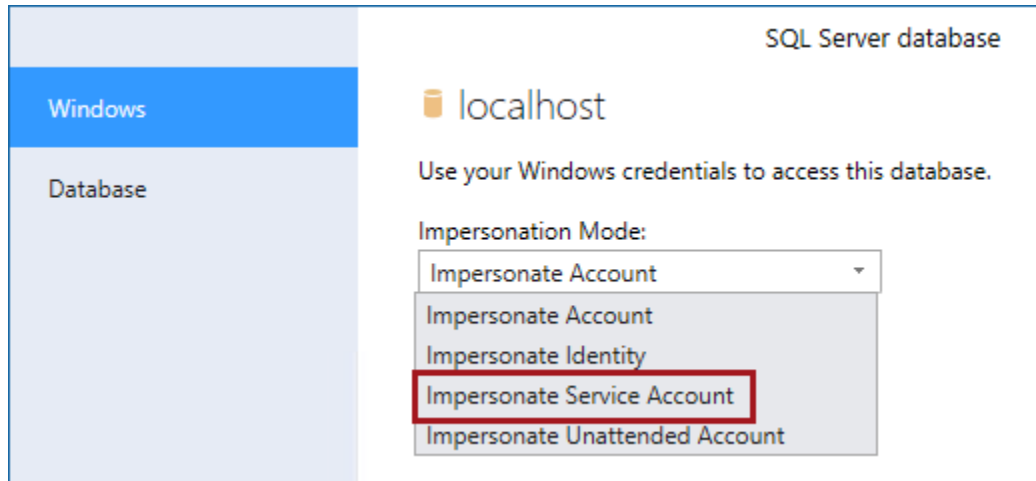

4. Click **Connect**.

5. In the **SQL Server Database** window, in the **Server** box, enter **localhost**.
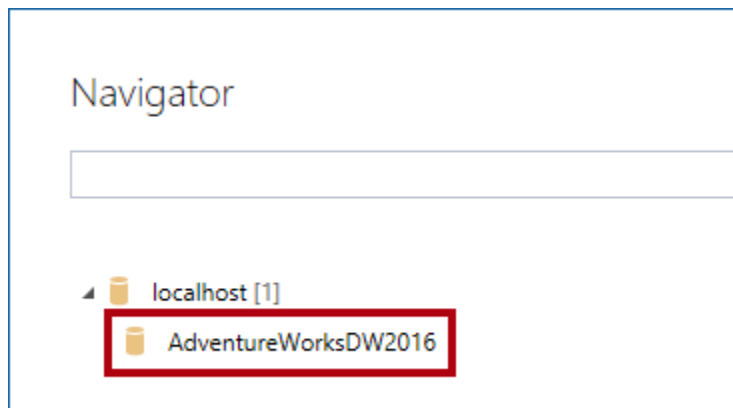

6. Click **OK**.

7. To configure the identity Analysis Services will use to authenticate with the database service, in the **Impersonation Mode** dropdown list, select **Impersonate Service Account**.
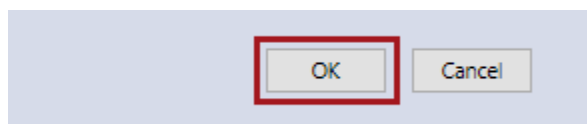

8. Click **Connect**.

9. When prompted to use an unencrypted connection, click **OK**.

10. In the **Navigator** window, select the **AdventureWorksDW2016** database.
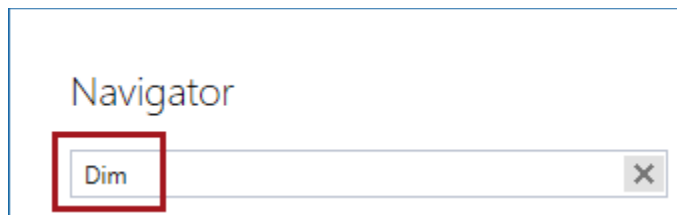

11. Click **OK**.

12. To filter table available objects, in the search box (located at the top-left), enter **Dim**.

    This search word will filter the available objects—tables and views—by those containing those sequence of letters. In this database, this reveals all dimension tables.

    Navigator

    Dim   ✕

13. Check the following four tables:

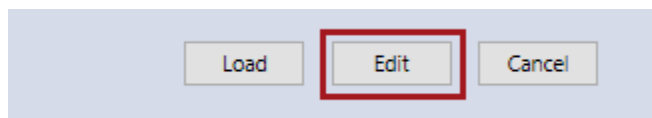 | Table Name |
    | --- |
    | DimDate |
    | DimEmployee |
    | DimProduct |
    | DimReseller |

14. Modify the value in the search box to **Fact**.

    Navigator

    Fact   ✕

15. Check the **FactResellerSales** table.

    Each selected table will generate a query, which can be refined to deliver a table to the model.

16. To design the queries for each selected table, click **Edit**.

    Load   Edit   Cancel

17. Notice that the **Power Query Editor** window opens.

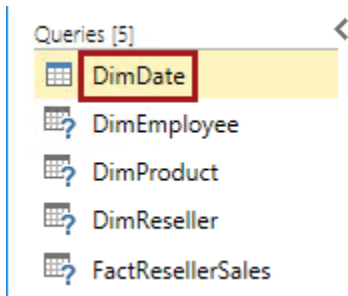   *This window is used to apply transformation steps to each query.*

18. To maximize the window, at the top-right corner, click the **Maximize** icon.

Configuring SQL Server Queries

In this task, you will configure each selected table as a query which will output a table to the model.

1. In the **Queries** pane (located at the left), notice that the first query—**DimDate**—is selected.
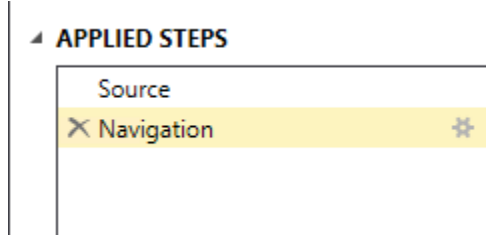

2. In the **Query Settings** pane (located at the right), in the **Name** box, replace the text with **Date**.
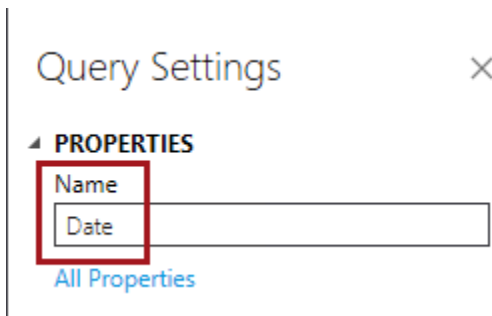

   The query name will become the table name, once loaded to the model. It is recommended that the name is concise and user-friendly.

3. In the **Applied Steps** pane, notice the two steps defined to retrieve all columns and rows from the **DimDate** table.
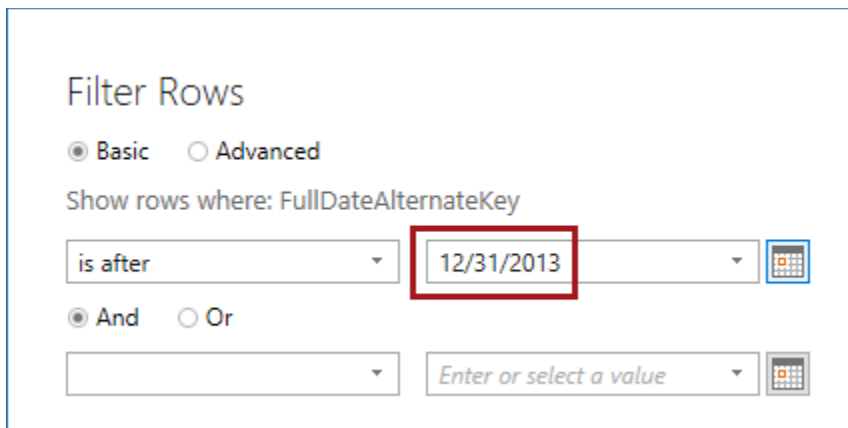

It is a common design approach to add steps to further transform the query, especially to filter the table by removing columns and/or rows, to add custom columns, or to apply more friendly names to columns.

4. To return a subset of rows from the source table, in the **FullDateAlternateKey** column header, click the down-arrow, and then select **Date Filters | After**.
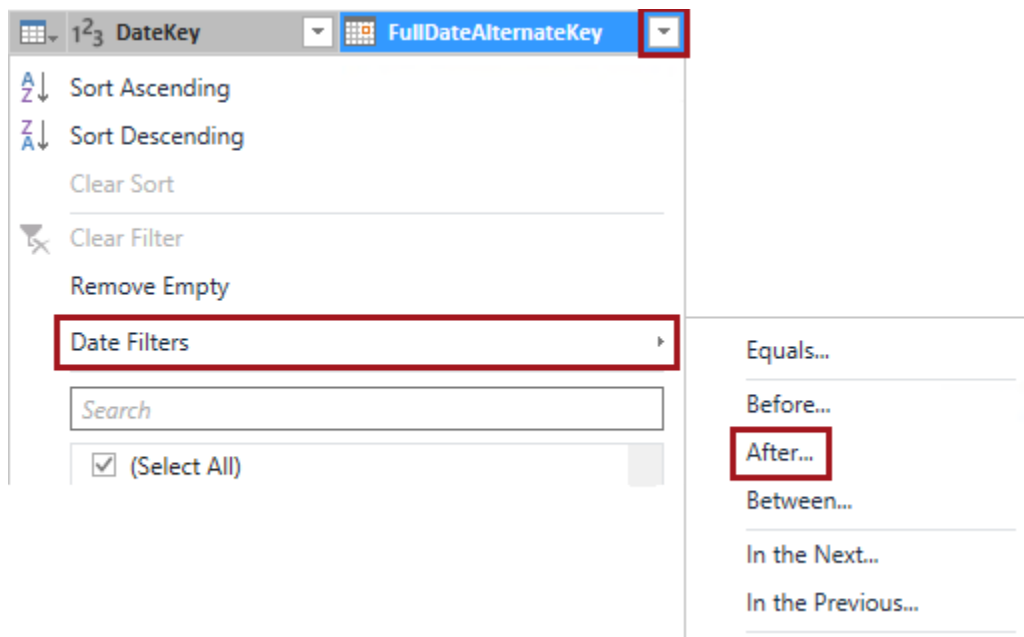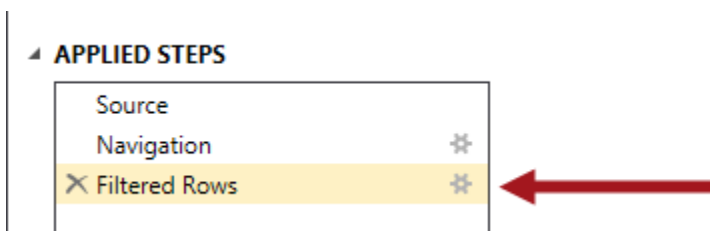
5. In the **Filter Rows** window, in the first box, enter **12/31/2013**.

    The VM regional settings are set to use United States, and so the date must be entered in MM/DD/YYYY format.
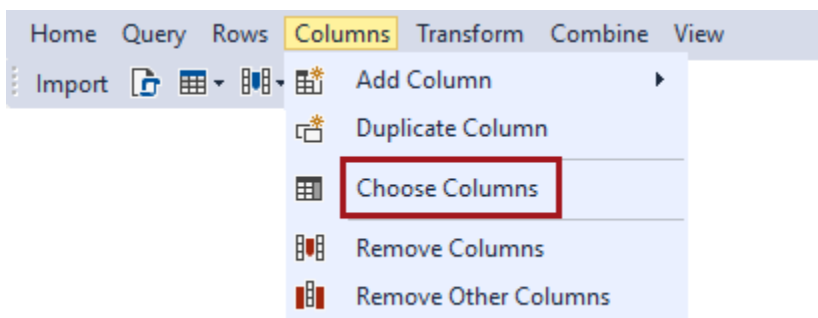

6. Click **OK**.

7. In the **Applied Steps** list, notice the addition of the **Filtered Rows** step.
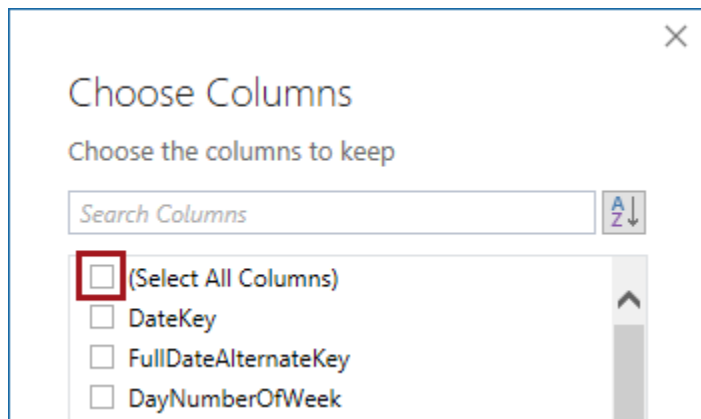

    Each transformation applied to the query is appended as a new step.

8. To retrieve a subset of columns, on the **Columns** menu, select **Choose Columns**.
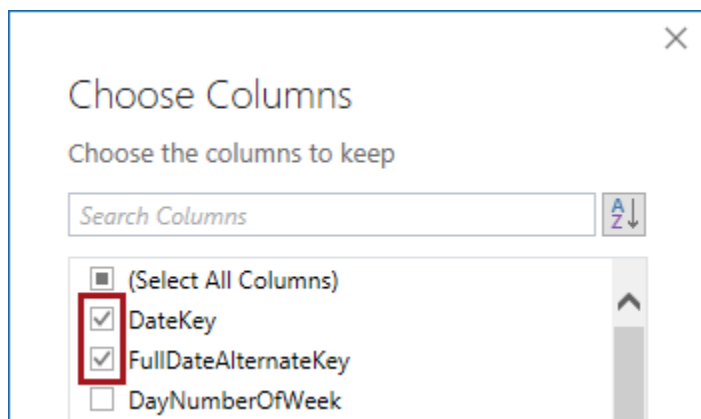

    Many of the menu commands may be more conveniently found on the toolbar commands directly beneath the menus.

9. In the **Choose Columns** window, to uncheck all columns, uncheck the first checkbox.
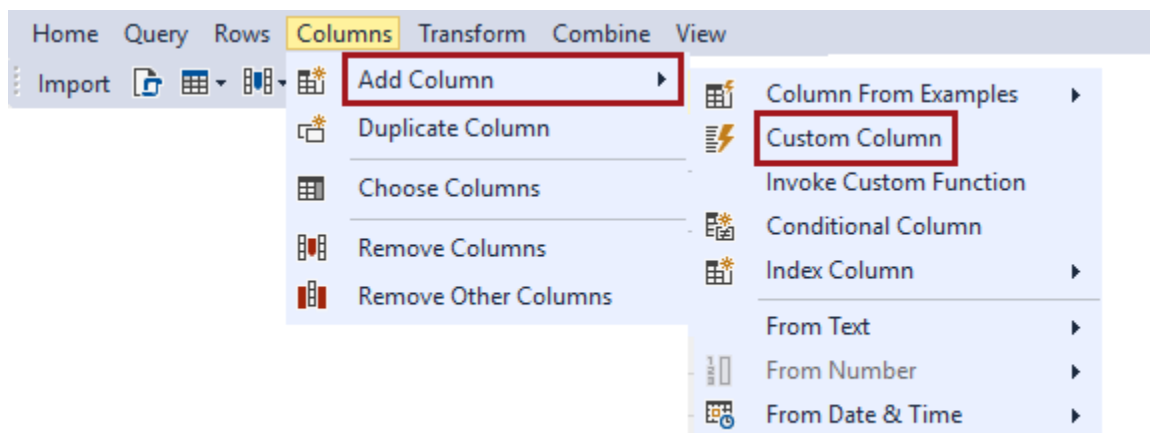

10. Check only the **DateKey** and **FullDateAlternateKey** columns.

11. Click **OK**.

12. To rename the second column, double-click the **FullDateAlternateKey** column header.

13. Replace the text with **Date**, and then press **Enter**.

14. To add a custom column, on the **Columns** menu, select **Add Column | Custom Column**.

15. In the **Custom Column** window, in the **New Column Name** box, replace the text with **Year**.
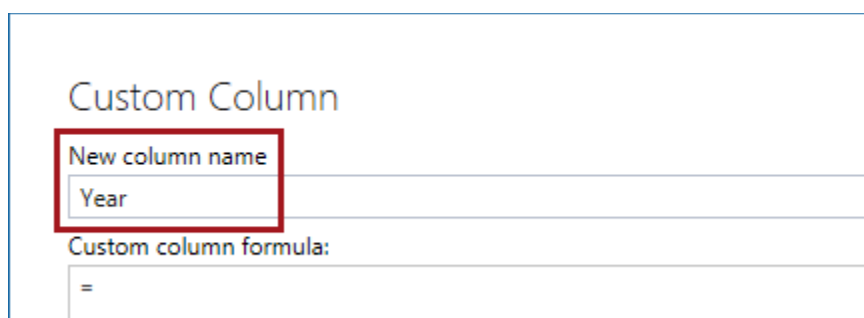
16. In the **Custom Column Formula**, enter the following expression.

    For convenience, all expressions in this exercise can be copied from the F:\Lab\Lab02\Assets\Snippets.txt file.

    Power Query (M) Formula Language
    ```
    "CY" & Number.ToText(Date.Year([Date]))
    ```

17. Click **OK**.

18. Notice the addition of a new step, and the new column.

19. Add the following two custom columns:

| Column Name	Expression
Quarter	`[Year] & " Q" & Number.ToText(Date.QuarterOfYear([Date]))`
Month	`Number.ToText(Date.Year([Date])) & " " & Date.ToText([Date], "MMM")`

20. In each column header, to the left of the column name, notice the icon which represents the column data type.

21. Notice that all the custom column data types are set to **ABC 123**.

    This icon represents "any" data type. It is, however, a better practice to explicitly set data types because once loaded as a table to the model, correct data types will be required to create relationships between tables, and also to produce calculations.
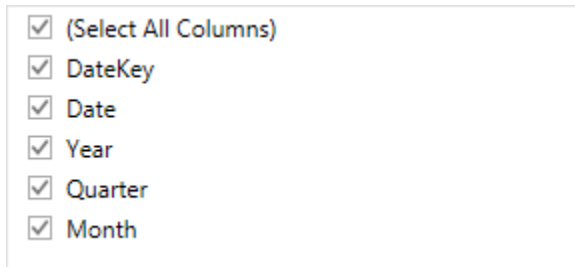
22. In the **Year** column header, click the data type icon, and then select **Text**.
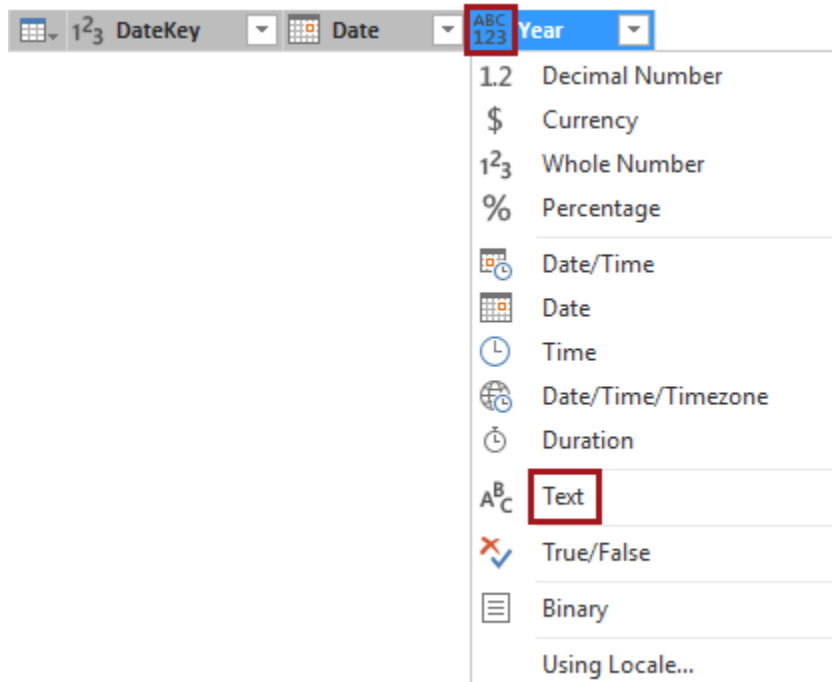
23. Update also the following column data types:

Column Name	Data Type
Quarter	Text
Month	Text

24. The validate the query columns, on the **Columns** menu, select **Choose Columns**.

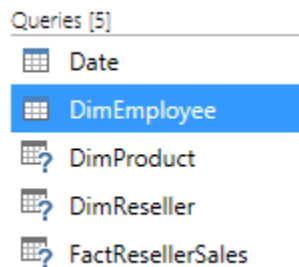25. Verify that the query consists of the following five columns:

It is important that you verify that the correct columns are defined. If the columns are different by name or number, you will need to review and revise the query design before continuing with the next steps.
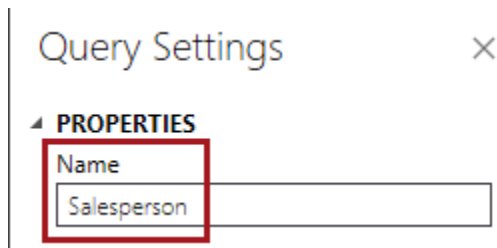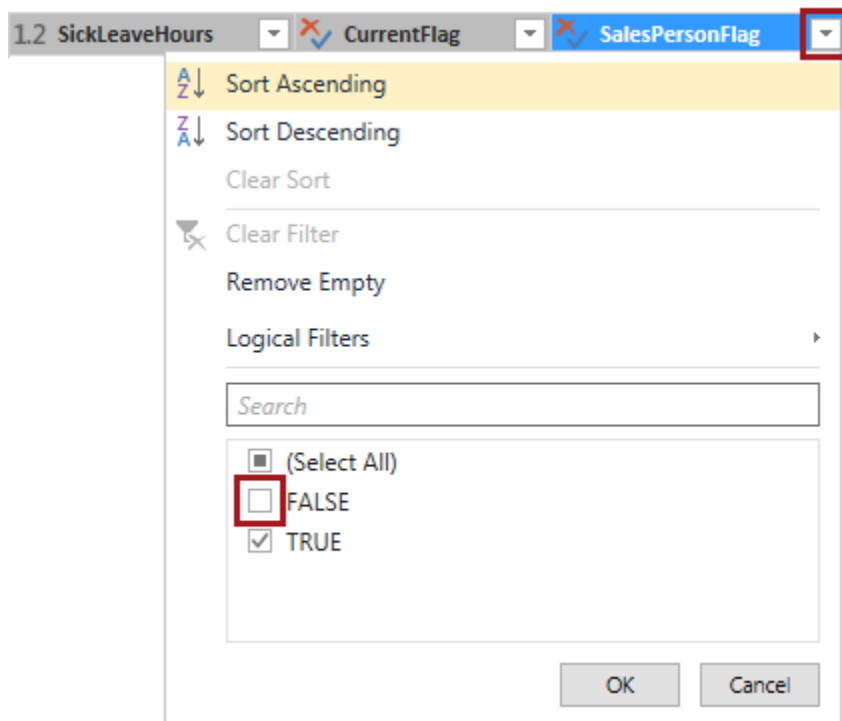
26. Click **Cancel**.
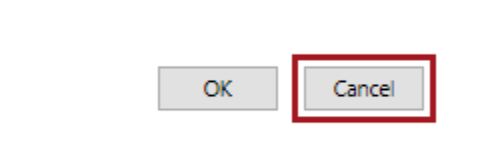
27. In the **Queries** pane, select the **DimEmployee** query.

28. Rename the query to **Salesperson**.

Query Settings   ✕

⊿ PROPERTIES
Name
Salesperson

29. In the **SalesPersonFlag** column header, click the down-arrow, and then apply a filter to
retrieve only those rows where the column value is **TRUE**.

| 1.2 SickLeaveHours | ▾ | CurrentFlag | ▾ | SalesPersonFlag | ▾ |

A↓ Sort Ascending
Z↓ Sort Descending
   Clear Sort
⊽ Clear Filter
   Remove Empty
   Logical Filters   ▸

Search

■ (Select All)
☐ FALSE
☑ TRUE

OK   Cancel

30. Add a custom column named **Salesperson**, using the following expression:

Power Query (M) Formula Language
```
[FirstName] & " " & [LastName]
```

31. Set the **Salesperson** column data type to **Text**.

32. Limit the query to output only the following five columns:

Column Name
EmployeeKey
EmployeeNationalIDAlternateKey
LoginID
DimSalesTerritory
Salesperson

33. Notice that the **DimSalesTerritory** column contains **Value** links.

A^B_C LoginID	DimSalesTerritory	A^B_C Salesperson
adventure-works\stephen0	Value	Stephen Jiang
adventure-works\brian3	Value	Brian Welcker
adventure-works\michael9	Value	Michael Blythe
adventure-works\linda3	Value	Linda Mitchell
adventure-works\jillian0	Value	Jillian Carson
adventure-works\garrett1	Value	Garrett Vargas
adventure-works\tsvi0	Value	Tsvi Reiter
	Value	Pamela Ansman-Wolfe
adventure-works\shu0	Value	Shu Ito
adventure-works\josé1	Value	José Saraiva
adventure-works\david8	Value	David Campbell
adventure-works\amy0	Value	Amy Alberts
adventure-works\jae0	Value	Jae Pak
adventure-works\ranjit0	Value	Ranjit Varkey Chudukatil
adventure-works\tete0	Value	Tete Mensa-Annan
adventure-works\syed0	Value	Syed Abbas
adventure-works\rachel0	Value	Rachel Valdez
adventure-works\lynn0	Value	Lynn Tsoflias

*These links represent values in a related table. The query editor has recognized that a foreign key column in the **DimEmployee** table references the **DimSalesTerritory** table. You will now configure the query to "expand" the references table to introduce related columns.*

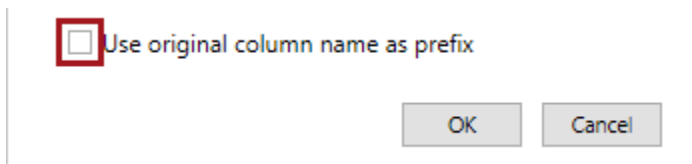34. In the **DimSalesTerritory** column header, click the **Expand** icon.

35. Uncheck the first checkbox, and then check the following three columns:

Column Name
SalesTerritoryRegion
SalesTerritoryCountry
SalesTerritoryGroup

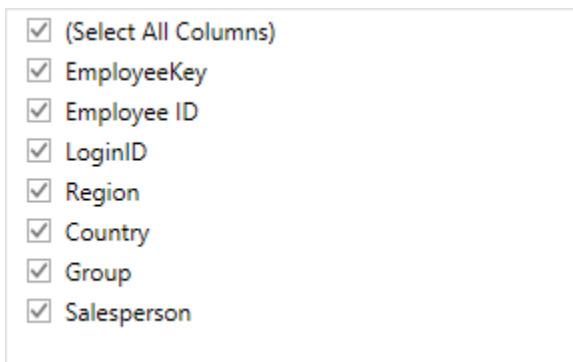36. At the bottom of the list, uncheck the **Use Original Column Name as Prefix** checkbox.

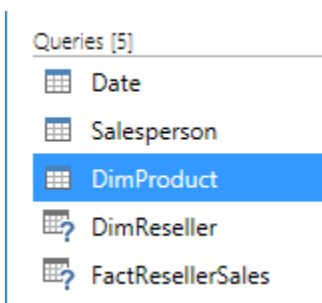

37. Click **OK**.

38. Rename the following four columns:

Column Name	New Column Name
EmployeeNationalIDAlternateKey	Employee ID
SalesTerritoryRegion	Region
SalesTerritoryCountry	Country
SalesTerritoryGroup	Group

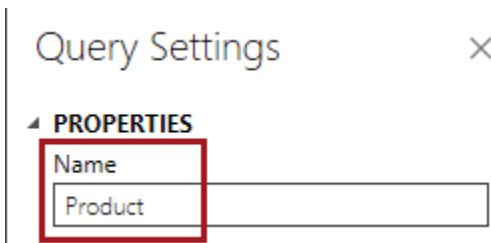39. Verify that the query consists of the following seven columns:

40. In the **Queries** pane, select the **DimProduct** query.

41. Rename the query as **Product**.

42. Filter the query to include only rows where the **FinishedGoodsFlag** column value is **TRUE**.

43. Limit the query to output only the following four columns:

Column Name
ProductKey
EnglishProductName
Color
DimProductSubcategory

44. Expand the **DimProductSubcategory** column to include only the following two columns:

Column Name
EnglishProductSubcategoryName
DimProductCategory

45. Expand the **DimProductCategory** column to include only the **EnglishProductCategoryName** column.

46. Rename the following three columns:

Column Name	New Column Name
EnglishProductName	Product
EnglishProductSubcategoryName	Subcategory
EnglishProductCategoryName	Category

47. Verify that the query consists of the following five columns:

48. In the **Queries** pane, select the **DimReseller** query.
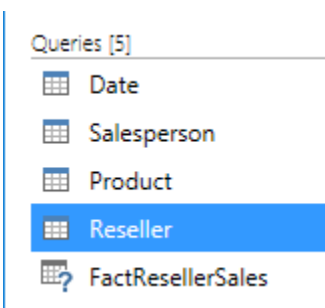
49. Rename the query as **Reseller**.
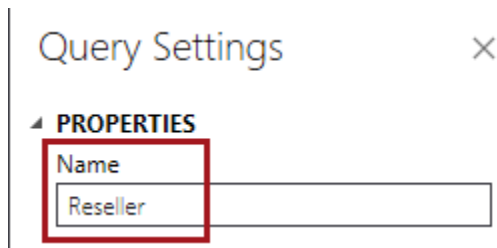
50. Limit the query to output only the following four columns:

Column Name
ResellerKey
BusinessType
ResellerName
DimGeography

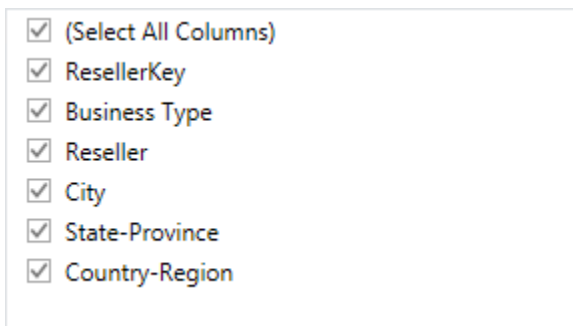51. Expand the **DimGeography** column to include only the following three columns:

Column Name
City
StateProvinceName
EnglishCountryRegionName

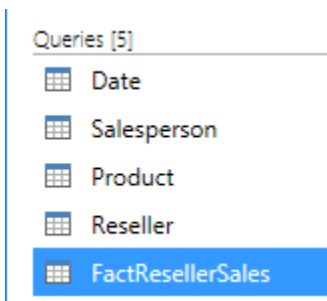52. Rename the following four columns:

Column Name	New Column Name
BusinessType	Business Type (include a space between the two words)
ResellerName	Reseller
StateProvinceName	State-Province
EnglishCountryRegionName	Country-Region

53. Verify that the query consists of the following six columns:

54. In the **Queries** pane, select the **FactResellerSales** query.

55. Rename the query as **Sales**.

56. Limit the query to output only the following seven columns:

Column Name
ProductKey
OrderDateKey
ResellerKey
EmployeeKey
OrderQuantity
TotalProductCost
SalesAmount

57. To import the queries into the model, on the toolbar, click **Import**.
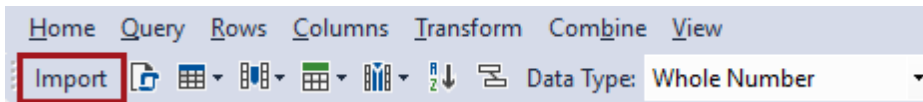
58. In the **Data Processing** window, verify that five tables were successfully loaded.

It is important that you verify that the correct number of rows were transferred. If the values are different, you will need to review, revise and re-import the queries.
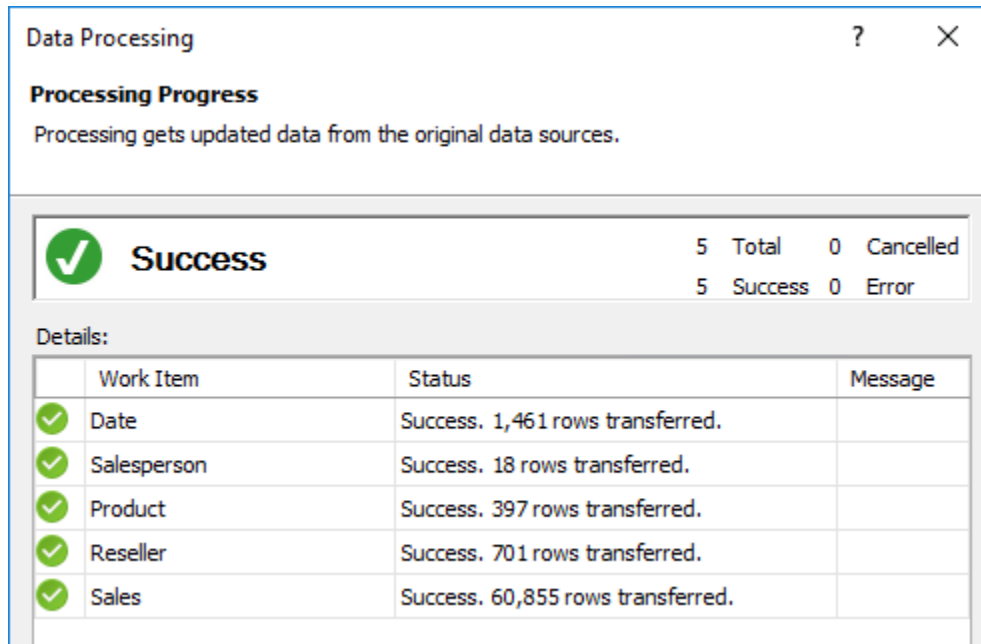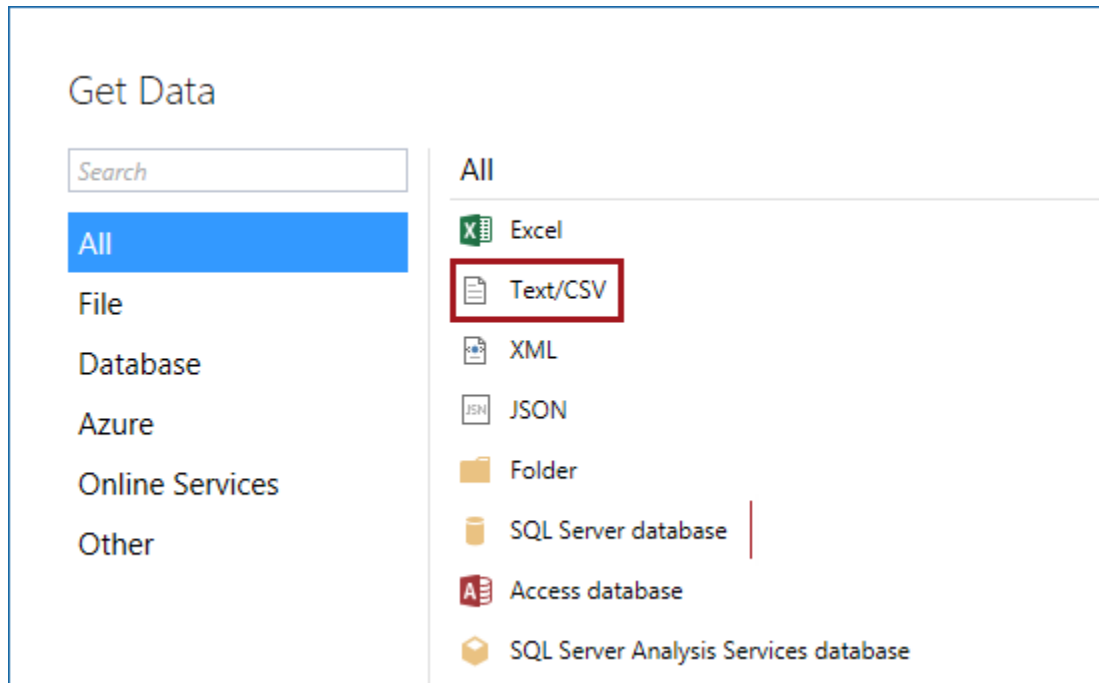

59. Click **Close**.

60. To save the project, on the **File** menu, select **Save All**.
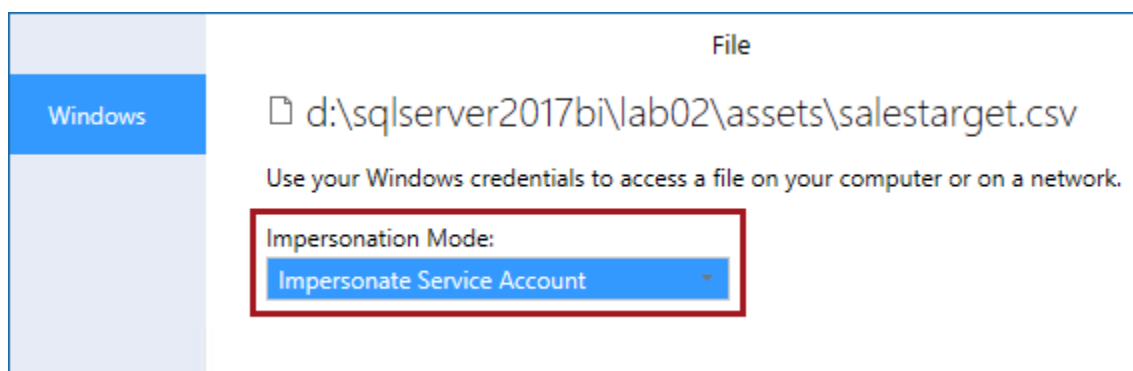
Creating a CSV Query

In this task, you will create a query to source data from a CSV file.

1. in **Tabular Model Explorer**, right-click the **Data Sources** folder, and then select **Import from Data Source**.

2. In the **Get Data** window, select the **Text/CSV** source.

3. Click **Connect**.

4. In the **Open** window, navigate to the **F:\Labs\Lab02\Assets** folder.

5. Select the **SalesTarget.csv** file.

6. Click **Open**.

7. In the **File** window, set the **Impersonation Mode** property to **Impersonate Service Account**.

8. Click **Connect**.

9. In the **SalesTarget.csv** window, review the preview data.

*The extract file contains one row per year per salesperson (by **Employee ID**), and quarterly target values are stored in the remaining columns.*

10. Click **OK**.

11. In the **Power Query Editor** window opens, in the **Query Settings** pane, in the **Name** box, replace the text with **Target**.

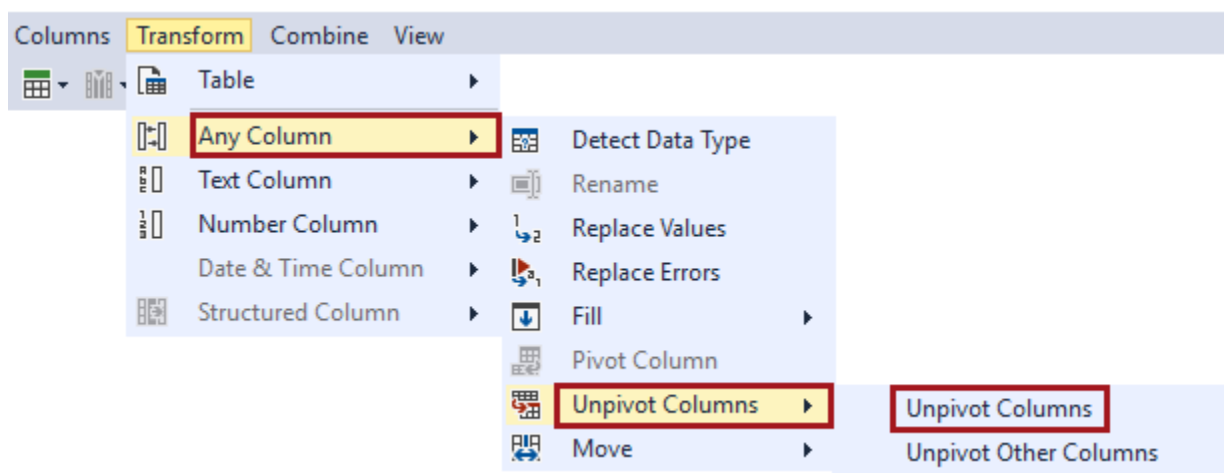12. To multi-select the quarter columns, first select the **Q1** column header, and then while pressing the shift-key, select the **Q4** column.

13. On the **Transform** menu, select **Any Column | Unpivot Columns | Unpivot Columns**.
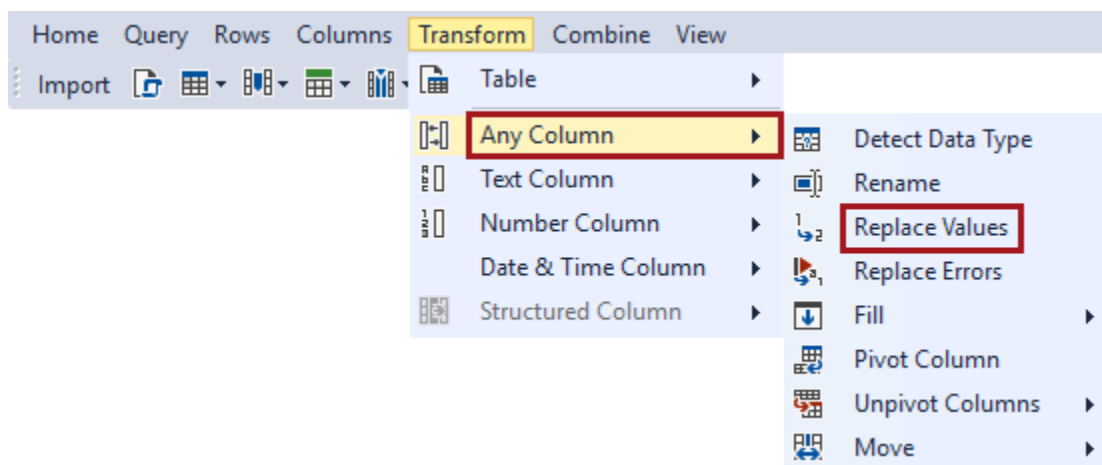

14. Rename the following two columns:

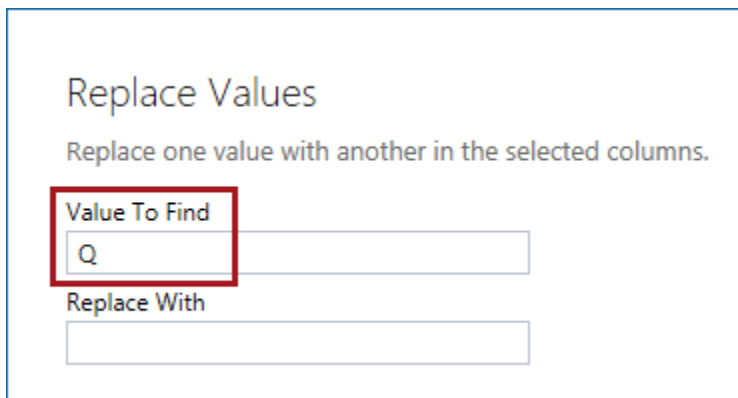| Column Name	New Column Name
Attribute	Quarter
Value	TargetAmount

15. Set the **TargetAmount** column data type to **Currency**.

16. Select the **Quarter** column.

17. On the **Transform** menu, select **Any Column | Replace Values**.

18. In the **Replace Values** window, in the **Value to Find** box, enter **Q**.

Replace Values

Replace one value with another in the selected columns.
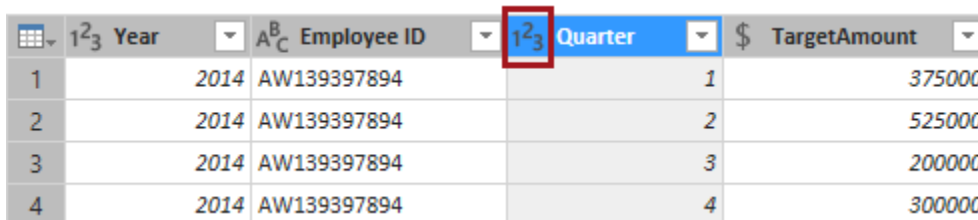
Value To Find

Q

Replace With

*This configuration will replace all instances of the letter **Q** with an empty string.*

19. Click **OK**.

OK   Cancel

20. Set the **Quarter** column data type to **Whole Number**.

1²₃ Year	A^Bc Employee ID	1²₃ Quarter	$ TargetAmount
1	2014	AW139397894	1	375000
2	2014	AW139397894	2	525000
3	2014	AW139397894	3	200000
4	2014	AW139397894	4	300000

21. Add a custom column named **DateKey**, by using the following expression:

Power Query (M) Formula Language

```
([Year] * 10000) + (((([Quarter] * 3) - 2) * 100) + 1
```

The expression determines the first date of the quarter, and produces the integer date value.

22. Set the **DateKey** column data type to **Whole Number**.

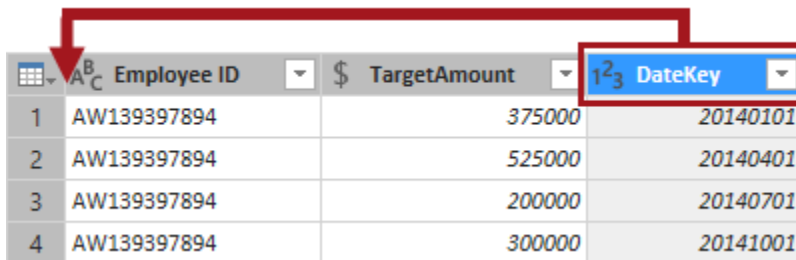23. Limit the query to output only the following three columns:
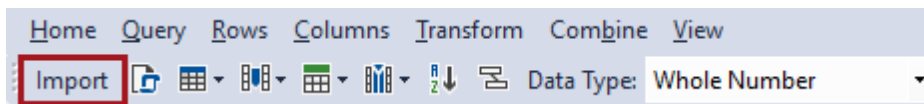
Column Name
Employee ID
TargetAmount
DateKey

24. To re-sequence the columns, drag the **DateKey** column header, and drop it to the left of the **Employee ID** column.
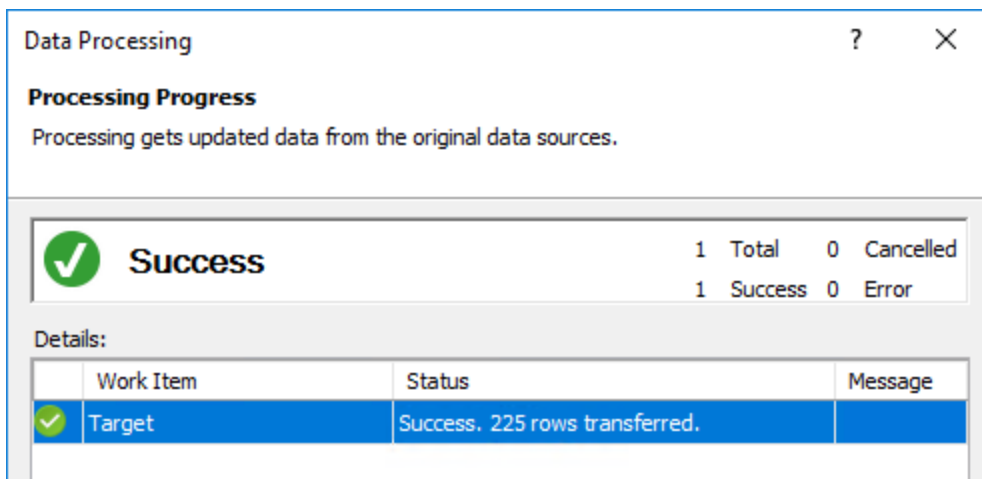

25. Import the query into the model.

26. In the **Data Processing** window, verify that the table was successfully loaded.

27. Click **Close**.

28. To save the project, on the **File** menu, select **Save All**.

Exercise 3: Configuring Relationships

In this exercise, you will explore the model to review the relationships created by the import process. You will also explore the **Sales** table, and then configure additional relationships.

Exploring the Model

In this task, you will explore the model tables in Diagram View.

1. To view the model in Diagram View, in **Tabular Model Explorer**, right-click the **Reseller Sales** model, and then select **Diagram View**.
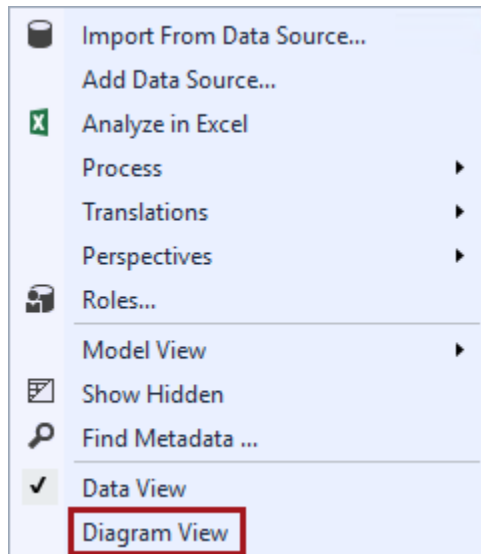
Tip: It is also possible to toggle between Data View and Diagram View by clicking the icons located at the bottom-right corner.


Diagram View is a useful way to visualize the model's tables and relationships. Beyond the model visualization, this view exposes useful functionality to enhance the design of the model. Oftentimes this functionality is available in Data View (the default view consisting of the tables and rows).

Note that calculated columns and measures can only be defined in Data View, and hierarchies can only be defined in Diagram View.

When appropriate, you can choose to work in the view that is most productive for you.

2. Reposition and resize each table so that the model is easy to understand.
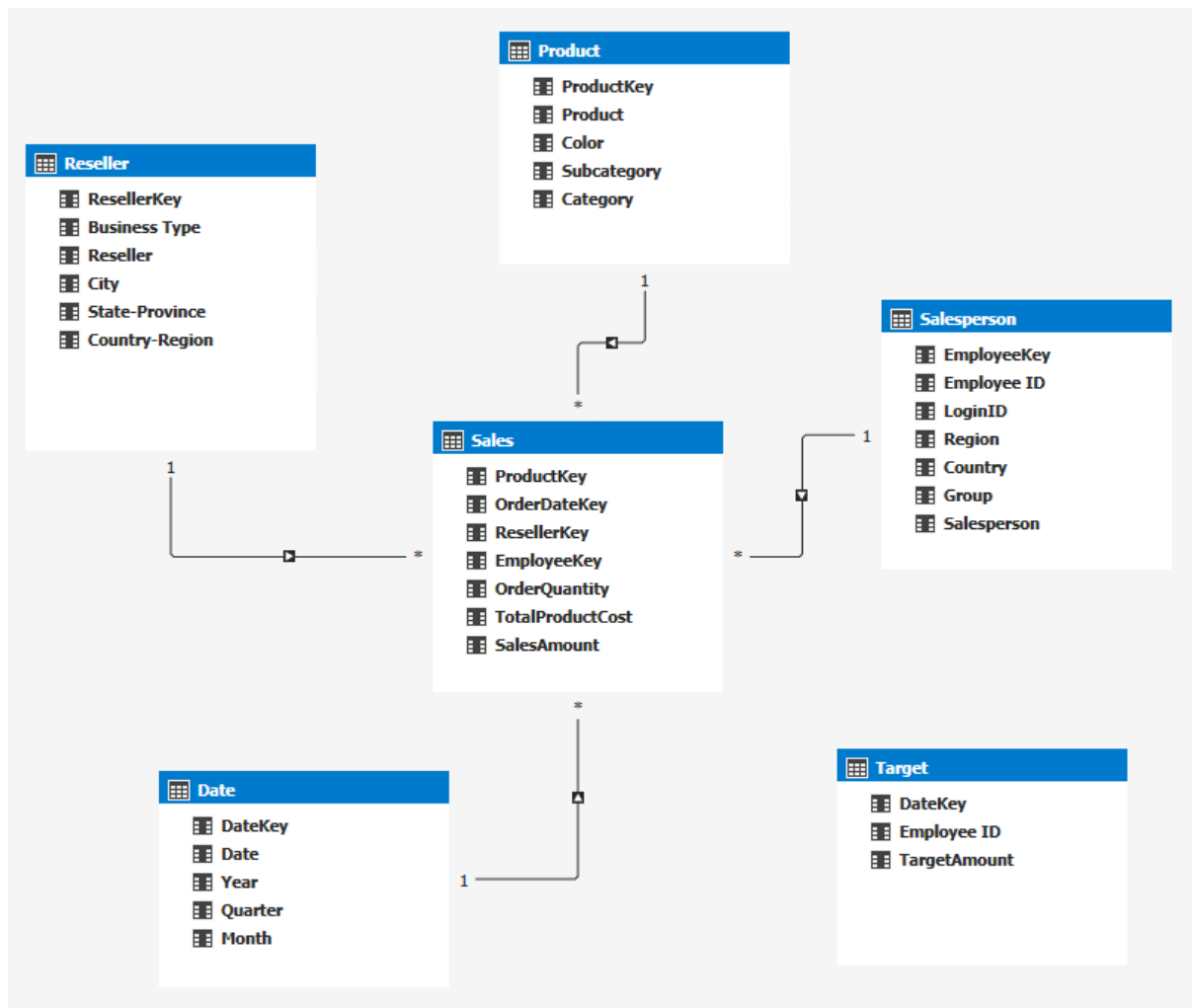
3. Notice that the **Target** table is not related to any table.

    Relationships may be automatically detected and added only when the queries are imported at the same time. Additionally, columns must be the same (or compatible) data types, and there must be a one-side to the relationship where the column values are unique.

4. To save the project, on the **File** menu, select **Save All**.

Exploring the Sales Table

In this task, you will explore the data loaded into the **Sales** table.

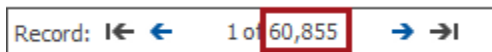1. In the diagram, right-click the **Sales** table, and then select **Go To**.
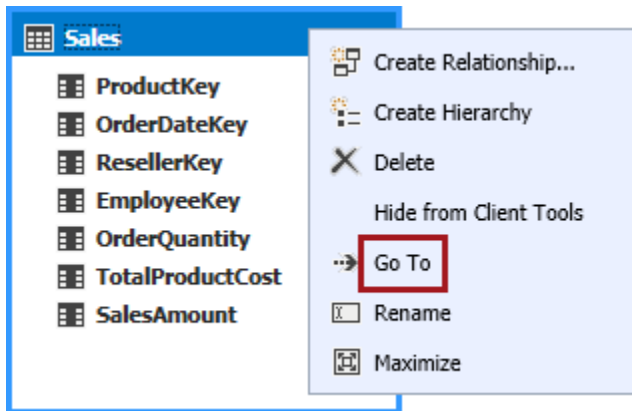
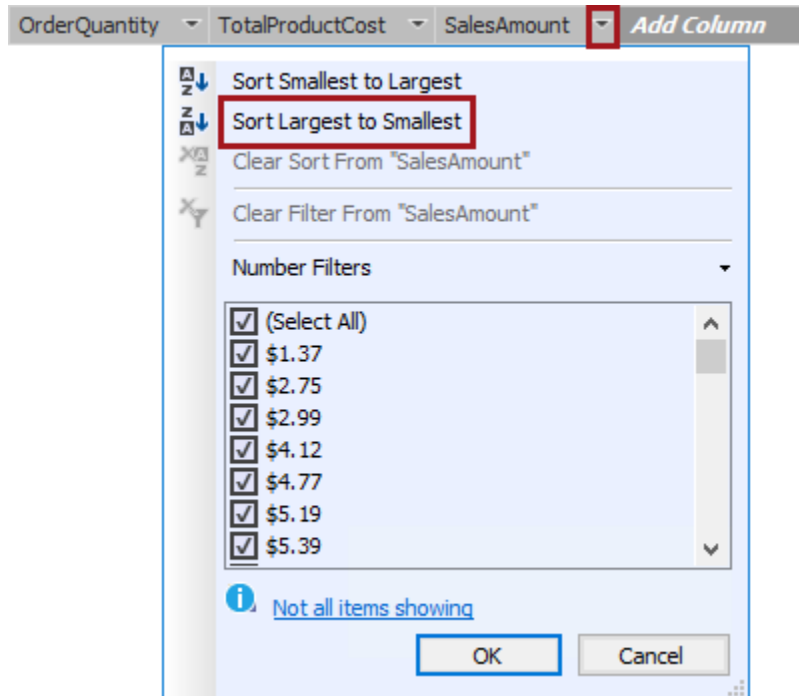*This switches the model designer to Data View.*

2. Along the bottom of the designer, notice that the **Sales** table is now in focus.
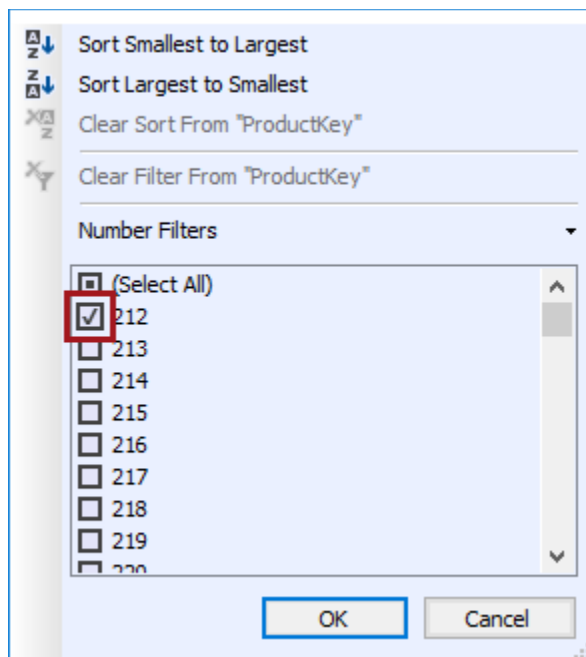

3. At the bottom-left corner, notice that this table has 60,855 rows.

4. To sort the rows by descending **SalesAmount** value, in the **SalesAmount** column header, click the down-arrow, and then select **Sort Largest to Smallest**.

5. To filter the rows by a specific product, in the **ProductKey** column header, click the down-arrow, uncheck **(Select All)**, and then check **212**.

6. Click **OK**.

---

7. Notice that the filtered table has 200 rows.

Record: |← ←   1 of 200   → →|

8. To reset the table sort, on the toolbar, click the **Clear Sort** button.

9. To remove all table filters, on the toolbar, click the **Clear All Filters** button.

It is important to understand that filtering and sorting data in Data View of the model designer does not alter the data already imported into the tables. It enables exploration of the data only.
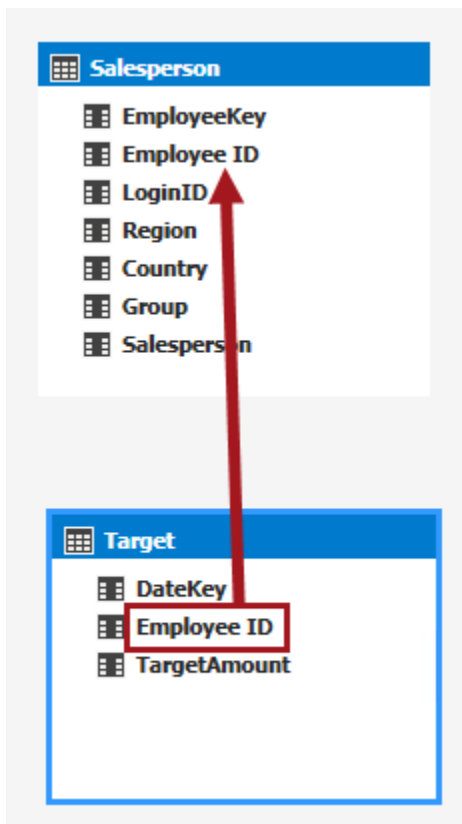
Adding Relationships

In this task, you will create a relationship between the **Target** and **Date** tables, and between the **Target** and **Salesperson** tables.

1. Switch to Diagram View.

2. To create a relationship between the **Target** and **Date** tables, drag the **DateKey** column onto the **DateKey** column of the **Date** table.
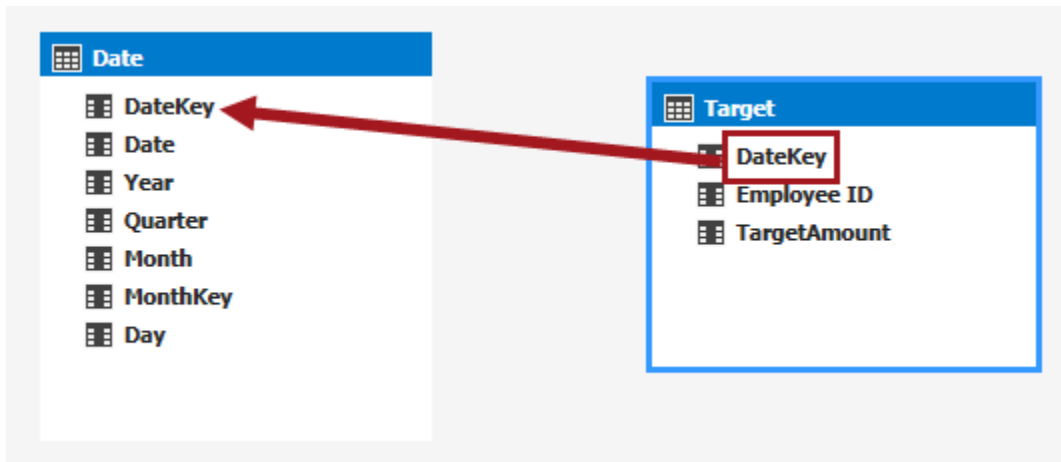
3. Create a second relationship, between the **Target** and **Salesperson** tables by dragging the **Employee ID** column onto the **Employee ID** column of the **Salesperson** table.

4. To review the model relationships, in **Tabular Model Explorer**, right-click the **Relationships** folder, and then select **Manage Relationships**.

5. Verify that there are six relationships listed.

*You can use the **Manage Relationships** window or the Diagram View to edit relationships and change their active status.*

6. Click **Close**.

7. To save the project, on the **File** menu, select **Save All**.

You have now completed the lab. In the next lab, you will enhance the model design by setting friendly column names, hiding column not intended for reporting, sorting column values, marking date tables, creating hierarchies, and adding calculation logic and a KPI.

If you are not immediately continuing with the next lab, you should complete the ***Finishing Up*** *exercise to shut down and stop the VM.*
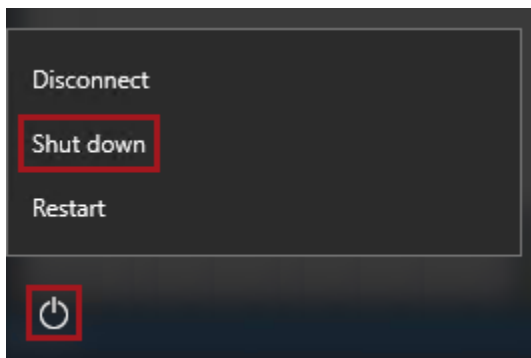
Finishing Up

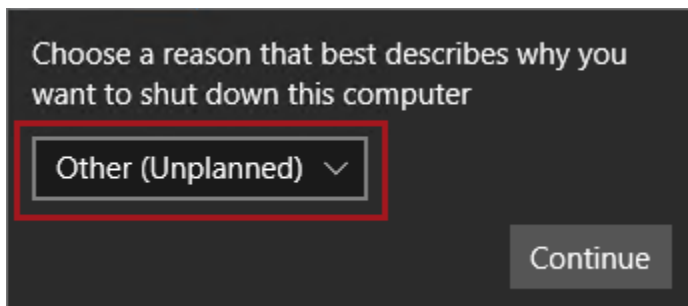In this exercise, you will shut down and stop the VM.

Finishing Up

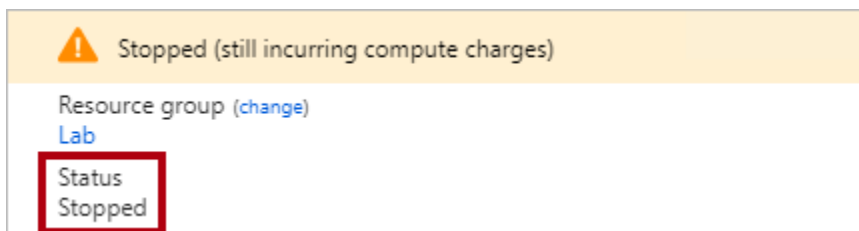In this task, you will shut down and stop the VM.

1. Close all open applications.

2. Press the **Windows** key, and then in the **Start** page, located at the bottom-left, click the **Power** button, and then select **Shut Down**.

    ```
    Disconnect

    Shut down

    Restart

    ⏻
    ```

3. When prompted to choose a reason, to accept the default.

    ```
    Choose a reason that best describes why you
    want to shut down this computer

    Other (Unplanned)  ∨

                                    Continue
    ```

4. Click **Continue**.

5. In the **Azure Portal** Web browser page, wait until the status of the VM updates to **Stopped**.

    ```
    ⚠ Stopped (still incurring compute charges)

    Resource group (change)
    Lab
    Status
    Stopped
    ```

   *In this state, however, the VM is still billable.*

---

6.   Optionally, to deallocate the VM, click **Stop**.

Deallocation will take some minutes to complete, and also extends the time required to restart the VM. Consider deallocating the VM if you want to reduce costs, or if you choose to complete the next lab after an extended period.
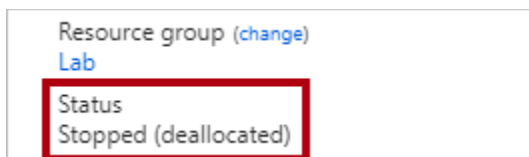

The deallocation can take several minutes to complete.

7.   When prompted to stop the virtual machine, click **Yes**.
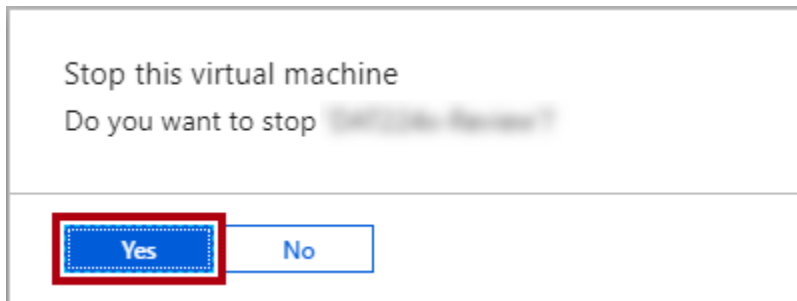
8.   Verify that the VM status updates to **Stopped (Deallocated)**.

In this state, the VM is now not billable—except for a relatively smaller storage cost.

Note that a deallocated VM will likely acquire a different IP address the next time it is started.

9.   Sign out of the **Azure Portal**.