## Working with Time Intelligence and KPIs

**Lab Time**: 60 minutes

**Lab Folder**: C:\Student\Modules\TimeIntelligenceAndKPIs\Lab

**Lab Overview**: In this module you will you will add a time dimension table into the data model and configure it so you can use it for financial analysis. Along the way you will also create a time dimension hierarchy and a KPI that monitors the growth in sales revenue.

### Exercise 1: Adding a Time Dimension Table

In this exercise you will import data from an external Excel workbook that you will use as a time dimension table.

#### Log in to Student Environment

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

#### Open Existing Workbook

1. Open the Excel workbook titled **WingtipSalesModel.xlsx** created in the previous lab.
   1. Navigate to the directory located in your student folder at **C:\Student\Models\**

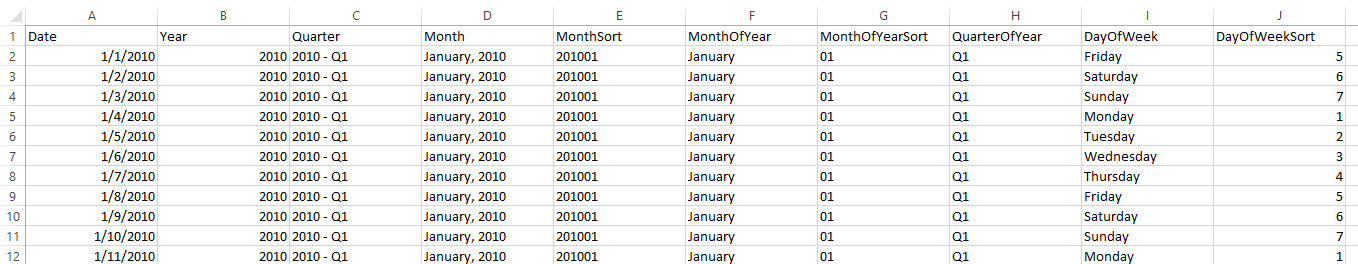
This lab assumes that you have completed the previous Power View lab in which you continued working on an Excel workbook named **WingtipSalesModel.xlsx**. If you would like to begin work on this lab without first completing the Power View lab, use the Windows Explorer to copy the lab solution file at **C:\Student\Modules\ExcelPowerView\Lab\Solution\WingtipSalesModel.xlsx** into the folder at **C:\Student\Models**.

* 1. Double-click on the Excel workbook file named **WingtipSalesModel.xlsx** to open it in Excel 2013.
  2. Once the file is open you may get prompted with a security warning. Click on the **Enable Content** button.



#### Import Data for a Time Dimension Table

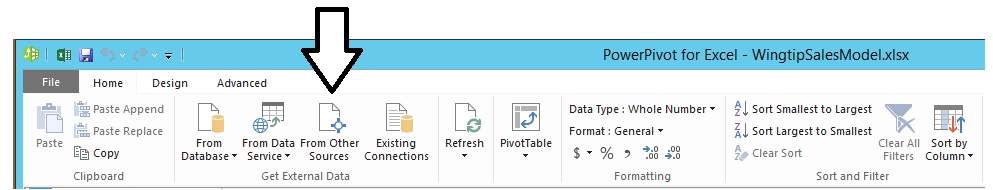
1. Inspect the data that you are about to import to create a time dimension table.
   1. The student lab files for this course include an Excel workbook named **WingtipTimeTable.xlsx** which contains a table with a row for each date between the 1/1/2010 and 12/31/2013. As you can see from the following screenshot, each row has a column for **Date**, **Year**, **Quarter** and **Month**. There are addition columns for the **MonthOfYear**, **QuarterOfYear** and **DayOfWeek** as well as sorting support columns named **MonthSort**, **MonthOfYearSort** and **DayOfWeekSort**.



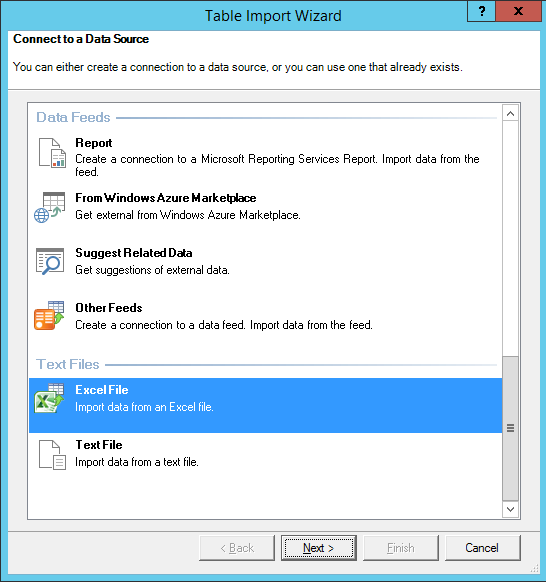
In this lab, you are not required to create a time table in Excel from scratch because the student lab files provide a pre-created time table for you. However, you might be curious (and you should be) as to how this time dimension table was created. The following table shows the Excel formulas that have been used to create in this time dimension table.

|  |  |
| --- | --- |
| Column | Value or Formula in Cell |
| **Date** | 1/1/2010 |
| **Year** | =YEAR(A2) |
| **Quarter** | =YEAR(A2) & " - " & CHOOSE(MONTH(A2), "Q1", "Q1", "Q1", "Q2", "Q2", "Q2", "Q3", "Q3", "Q3", "Q4", "Q4", "Q4") |
| **Month** | =TEXT(A2, "MMMM, YYYY") |
| **MonthSort** | =TEXT(A2, "YYYYMM") |
| **MonthOfYear** | =TEXT(A2,"MMMM") |
| **MonthOfYearSort** | =RIGHT("0" & MONTH(A2),2) |
| **QuarterOfYear** | =CHOOSE(MONTH(A2), "Q1", "Q1", "Q1", "Q2", "Q2", "Q2", "Q3", "Q3", "Q3", "Q4", "Q4", "Q4") |
| **DayOfWeek** | =TEXT(A2,"dddd") |
| **DayOfWeekSort** | =WEEKDAY(A2, 2) |

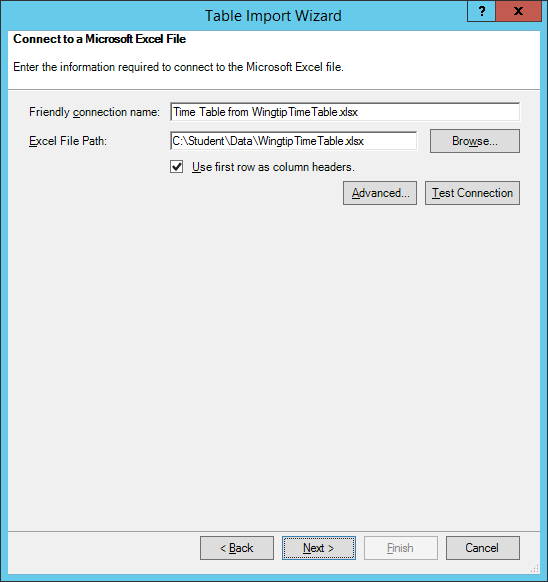
1. Import the time dimension table into the data model from an external Excel workbook.
   1. Navigate to the PowerPivot window for **WingtipSalesModel.xlsx**.
   2. In the **Get External Data** ribbon group, click the **From Other Sources** button to start the **Table Import Wizard**.



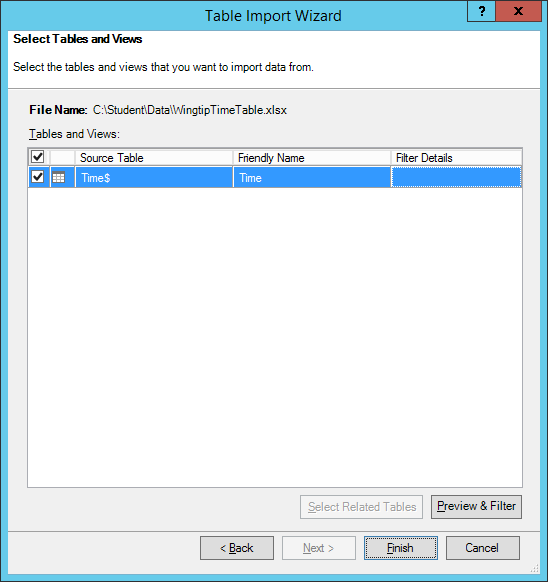
* 1. In the **Connect to a Data Source** page of the **Table Import Wizard**, scroll down the list of available sources, select **Excel file** and then click the **Next** button to move to the next page.



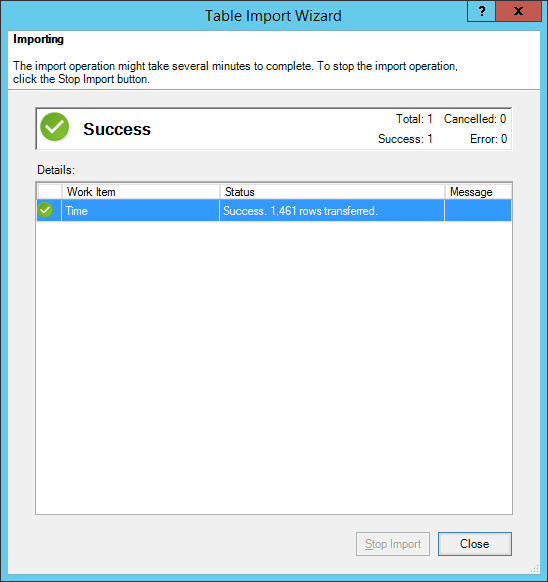
* 1. On the **Connect to a Microsoft Excel File** page, enter the following information:
     1. **Friendly connection name**: Time Table from WingtipTimeTable.xlsx
     2. **Excel File Path**: C:\Student\Data\WingtipTimeTable.xlsx
     3. **Use first row as column headers**: Checked



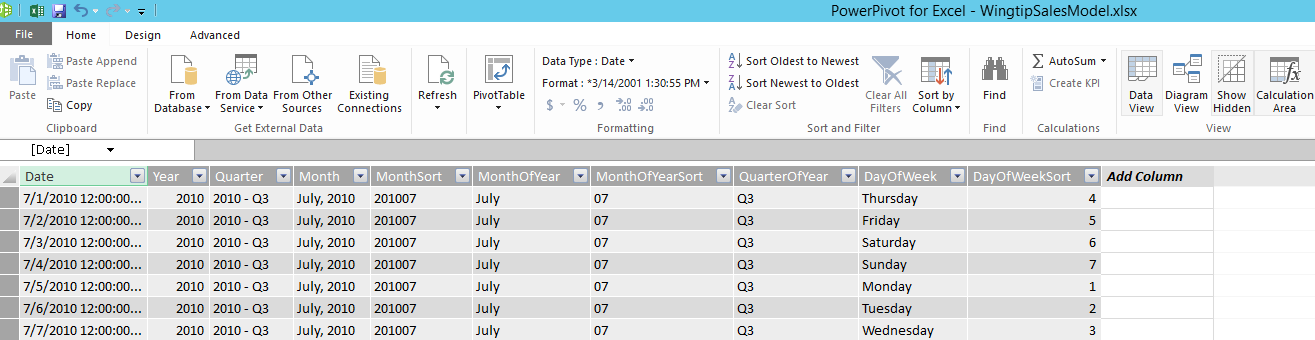
* 1. Click the **Next** button on the **Connect to a Microsoft Excel File** page to move to the **Select Tables and Views** page.
  2. On the **Select Tables and Views** page, accept the default settings and click the **Finish** button to begin the import process.



* 1. Once the import process completes, you should see a success confirmation which tells you that 1,461 rows have been imported which represents the number of days in the time period that runs from 1/1/2010 to 12/31/2013.

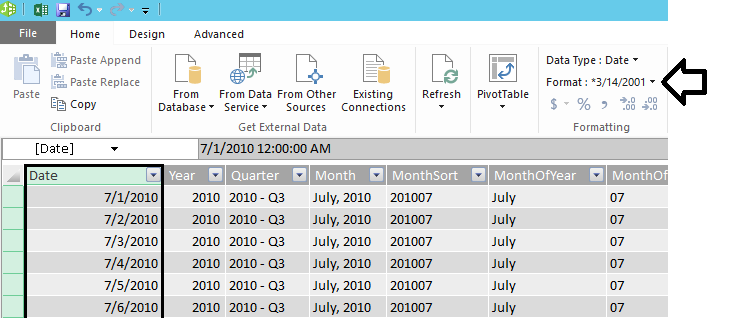


* 1. Click the **Close** button to close the **Table Import Wizard** and inspect the newly imported **Time** table. You should now be able to see the rows of the Time table inside the PowerPivot window.

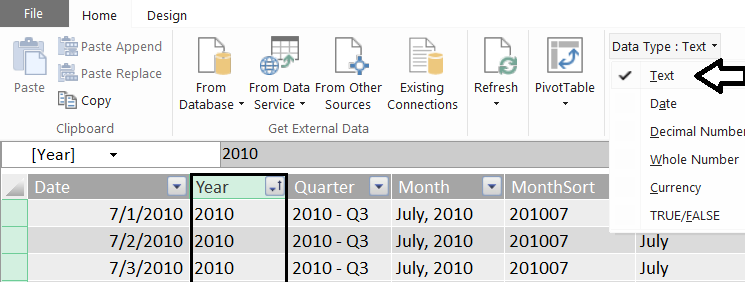


#### Configure Time Table Columns for Formatting and Sorting

1. In the Time table, select the **Date** column and adjust its formatting to a setting of **\*3/14/2001** to display the date without the time.

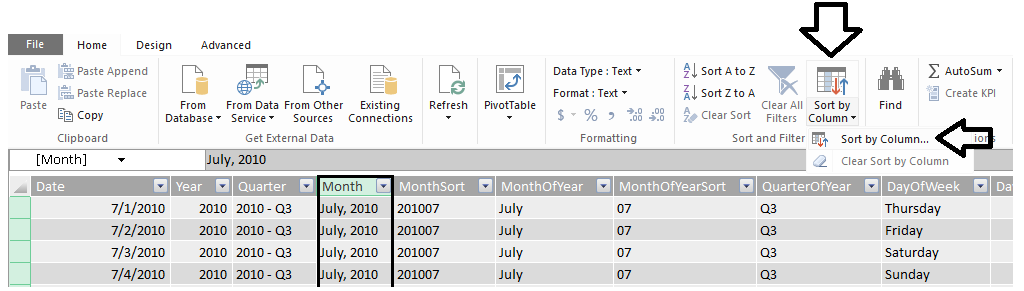


1. Configure the **Data Type** of the **Year** column to **Text** using the dropdown menu in the ribbon.

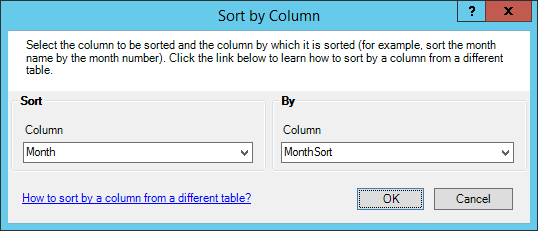


You might be wondering why it is important to modify the data type of the **Year** column to be a **Text** column. If you leave the **Data Type** setting at its default value of **Auto**, the column type is automatically set to the type **Whole Number**. This can cause confusion because the behavior in Excel when adding the column to a PivotTable assumes that the **Year** column values are numeric values that can be aggregated using operations such as SUM or AVERAGE. Configuring the **Year** column with the **Text** data type is a technique that prevents Excel from incorrectly assuming that it can aggregate values from the **Year** column.

1. Configure a custom sort column for the **Month** column.
   1. Select the **Month** column and then click the **Sort By Column** menu button in the ribbon and click the **Sort by Column**.

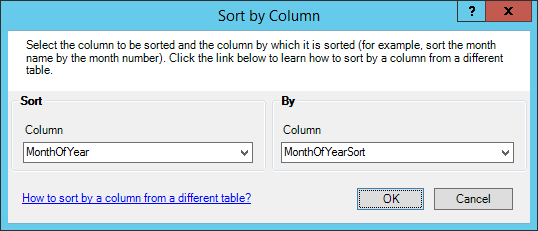


* 1. When the **Sort by Column** dialog appears, configure the **Month** column to be sorted by the **MonthSort** column as shown in the following screenshot.



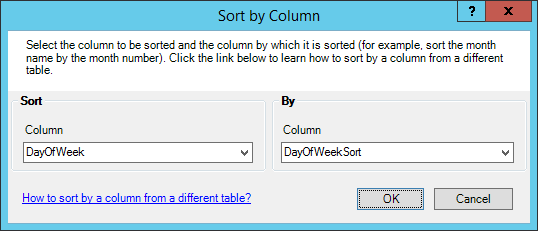
* 1. Click **OK** to save your changes.

1. Configure a custom sort column for the **MonthOfYear** column.
   1. Select the **MonthOfYear** column and then click the **Sort By Column** menu button in the ribbon and click the **Sort by Column**.
   2. When the **Sort by Column** dialog appears, configure the **MonthOfYear** column to be sorted by the **MonthOfYearSort** column as shown in the following screenshot.



* 1. Click **OK** to save your changes.

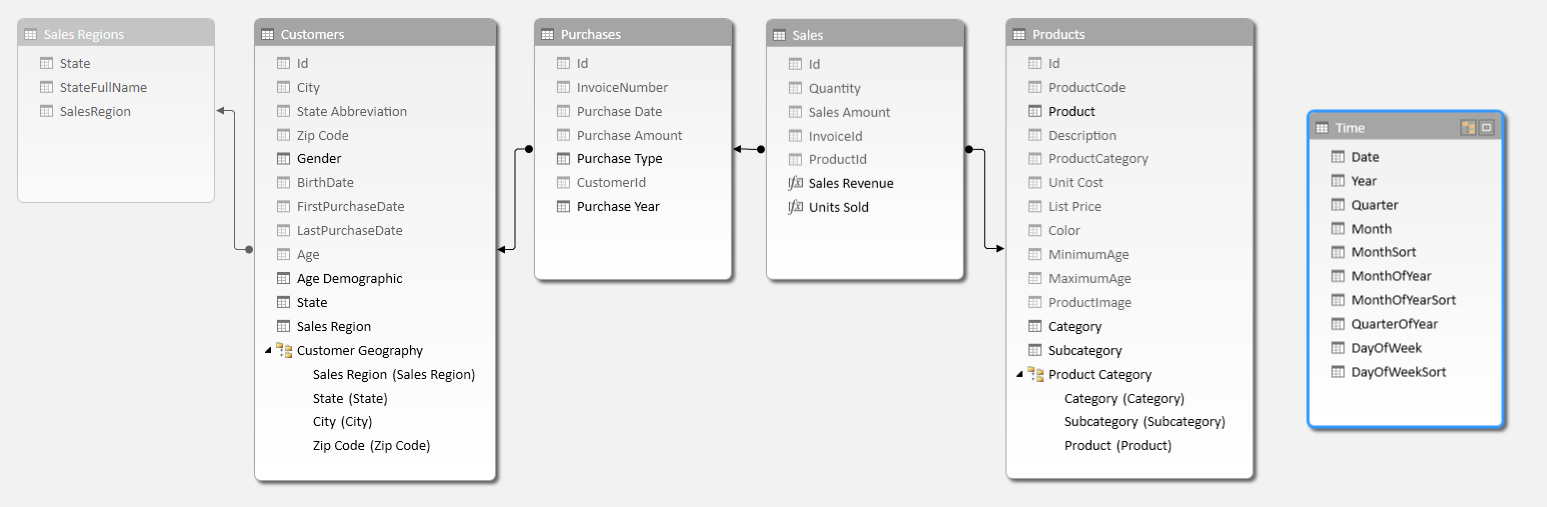
1. Configure a custom sort column for the **DayOfWeek** column.
   1. Select the **DayOfWeek** column and then click the **Sort By Column** menu button in the ribbon and click the **Sort by Column**.
   2. When the **Sort by Column** dialog appears, configure the **DayOfWeek** column to be sorted by the **DayOfWeekSort** column as shown in the following screenshot.



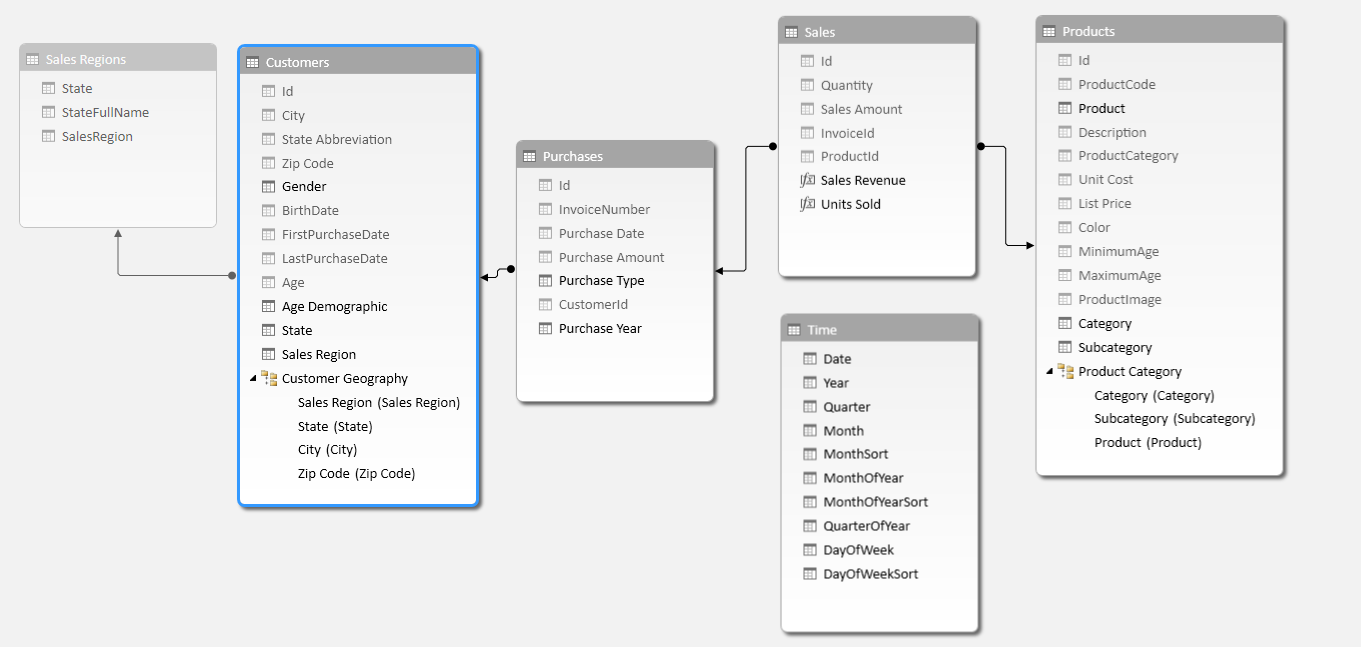
* 1. Click **OK** to save your changes.

#### Define a Relationship between the Purchases table and the Time table

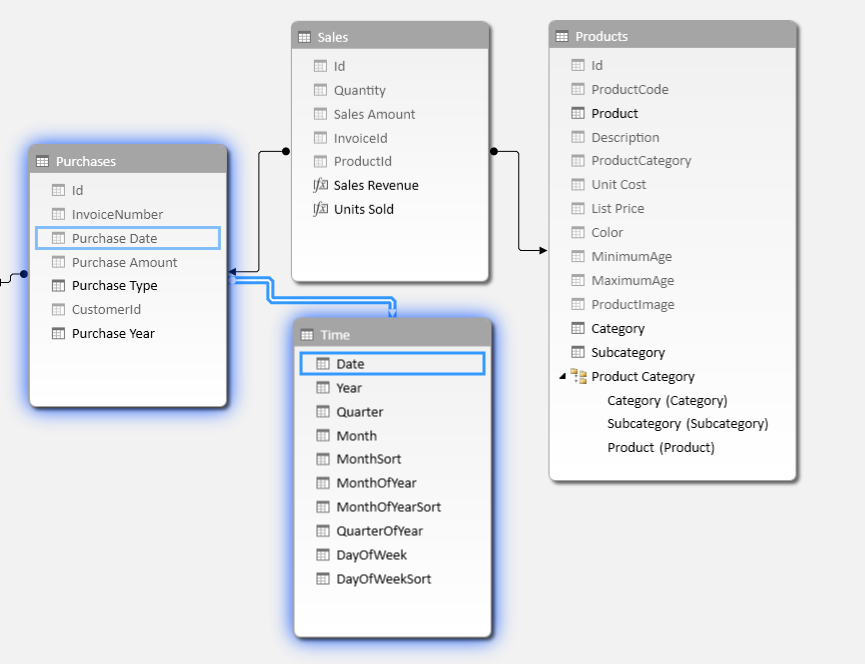
1. Define a relationship between the **Purchases** table and the **Time** table.
   1. Switch the PowerPivot window over to Diagram View by clicking the **Diagram View** button in the ribbon.
   2. You should be able to see the **Time**. However, the **Time** table has been created without any relationships.



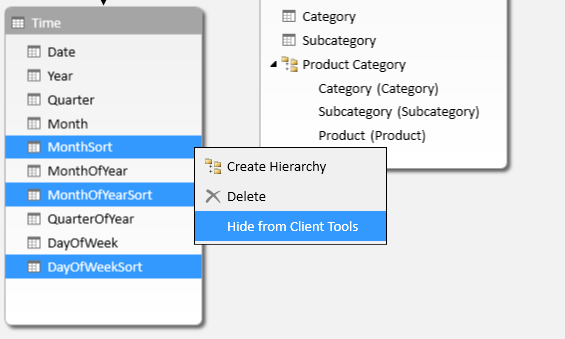
* 1. Using your mouse in Diagram View, rearrange the tables in diagram view so the layout appears as it does in the following screenshot with the **Time** table beneath the **Sales** table and just to the right of the **Purchases** table.



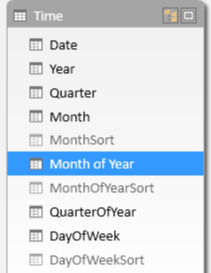
* 1. Using your mouse, drag-and-drop the **Purchase Date** column from the **Purchases** table on top of the **Date** column of the **Time** table to create a relationship between the two.



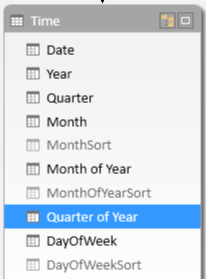
1. Hide the columns in the **Time** table named **MonthSort**, **MonthOfYearSort** and **DayOfWeekSort**.



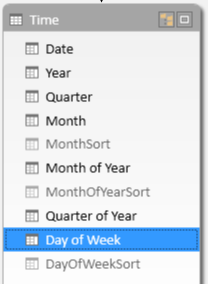
1. Rename the **MonthOfYear** column to **Month of Year** to make it more user-friendly.



1. Rename the **QuarterOfYear** column to **Quarter of Year** to make it more user-friendly.



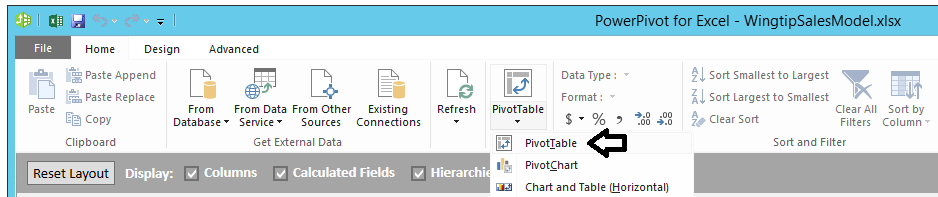
1. Rename the **DayOfWeek** column to **Day of Week** to make it more user-friendly.



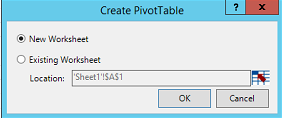
You have now configured the **Time** table to the point where you can give it a test drive by creating a PivotTable.

#### Create a PivotTable to Display Sales Revenue by Time Period

1. Create a new PivotTable that makes use of the **Time** table you just created.
   1. From the **Home** tab in the ribbon of the **POWERPIVOT** window, drop down the **PivotTable** menu control and click **PivotTable** to create a new pivot table in in the workbook **WingtipSalesModel.xslx**.



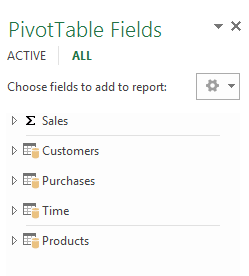
* 1. When the **Create PivotTable** dialog appears, select **New Worksheet** and click the **OK** button.



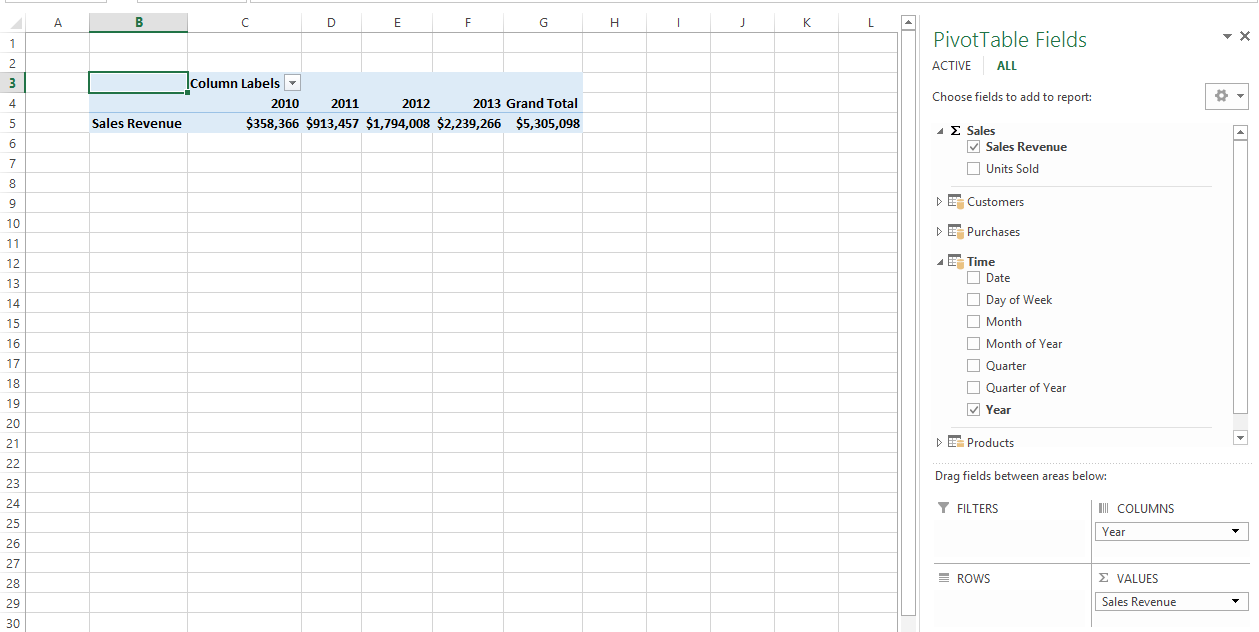
* 1. Once the new worksheet with the new PivotTable is added to the workbook in the Excel application window, rename the worksheet with a name of **Sales Revenue by Time Period**.



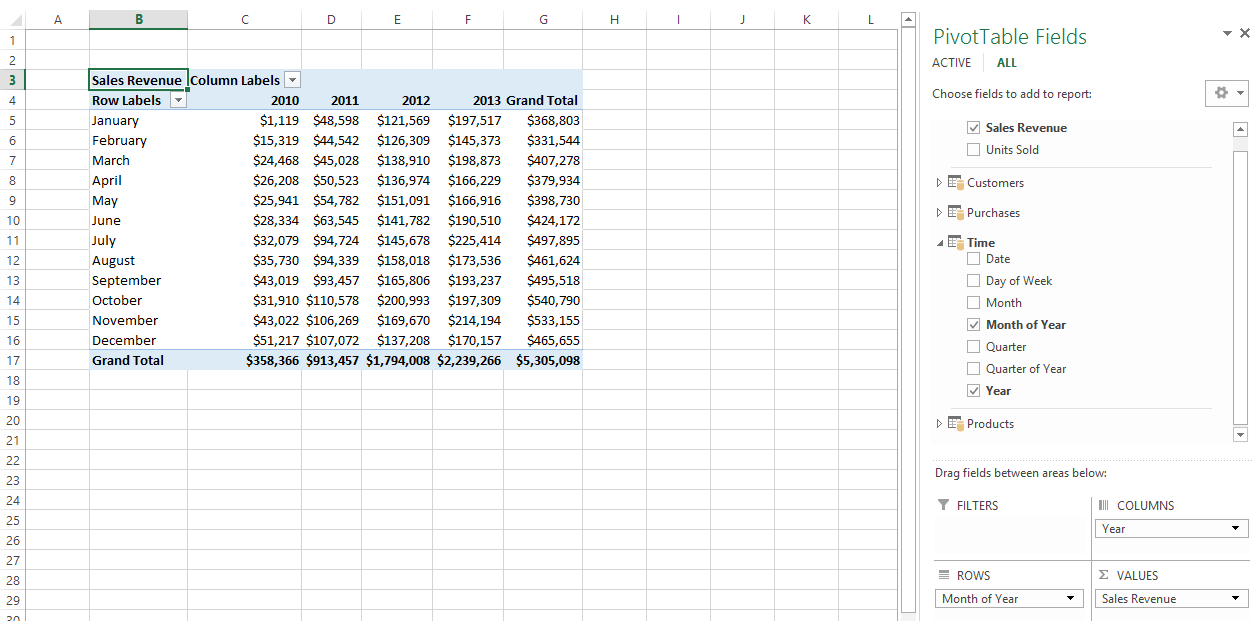
1. When you begin working on the new PivotTable, you should notice that the **Sales** table is now in the **PivotTable Fields** section.



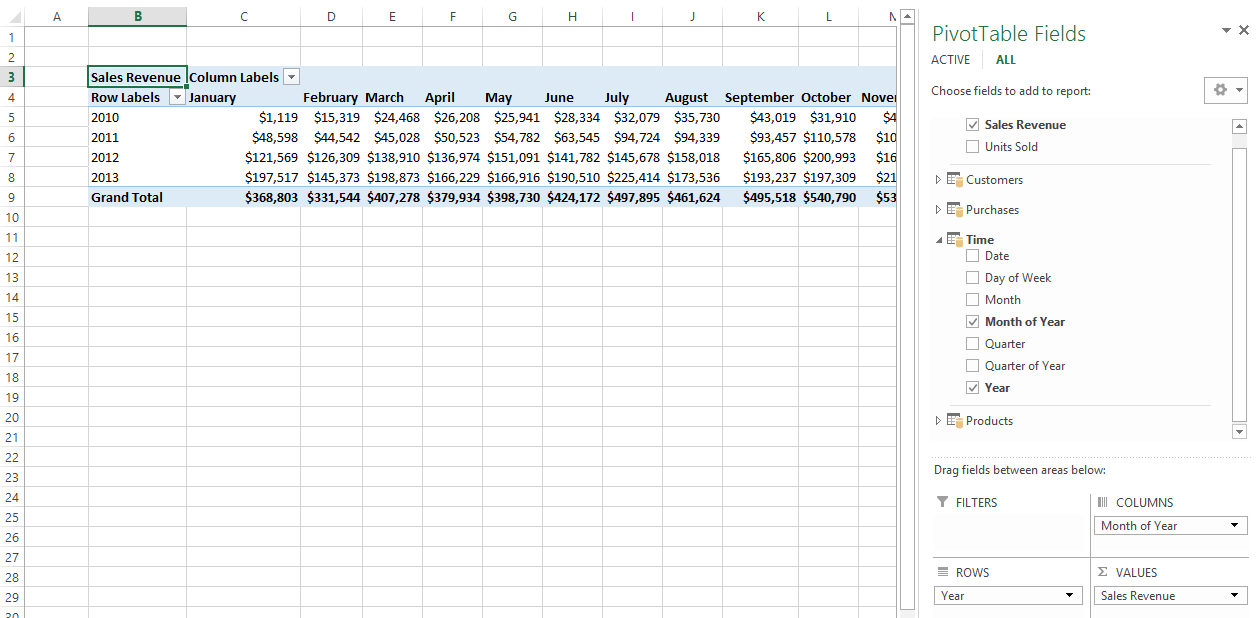
1. Create new pivot table by adding the **Sales Revenue** calculated fields from the **Sales** table to the **VALUES** section and the **Year** column of the **Time** table to the **COLUMNS** section as shown in the following screenshot.



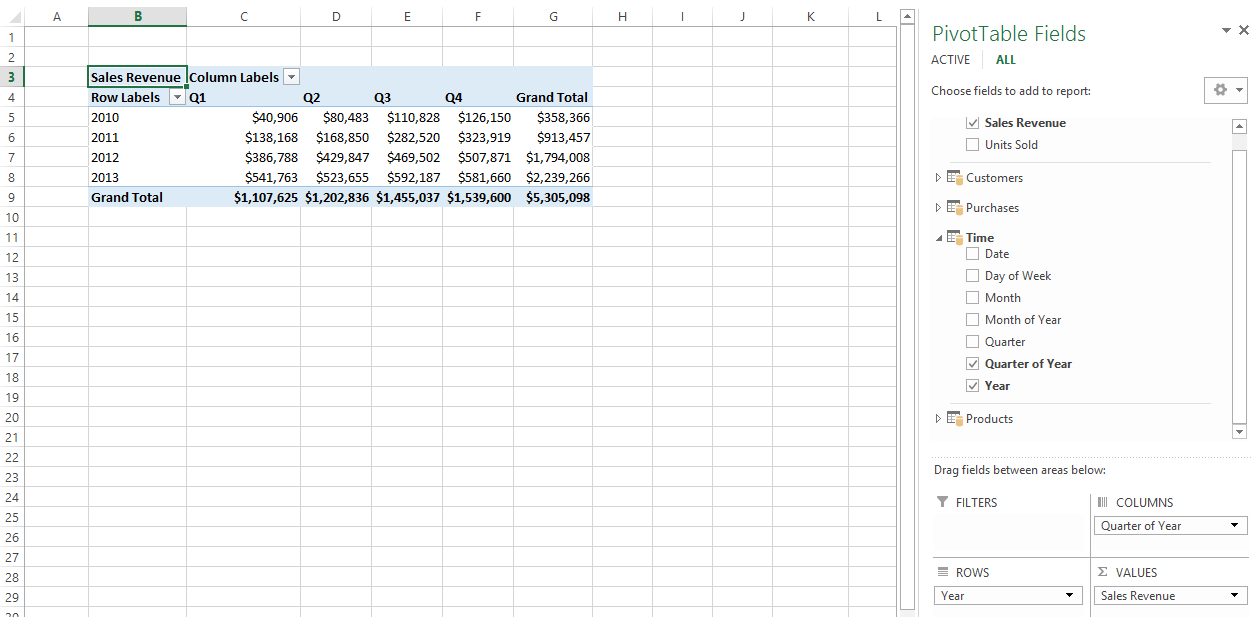
1. Now add **Month of Year** column from the **Time** table into the **ROWS** section. Your PivotTable should appear as the one shown in the following screenshot.



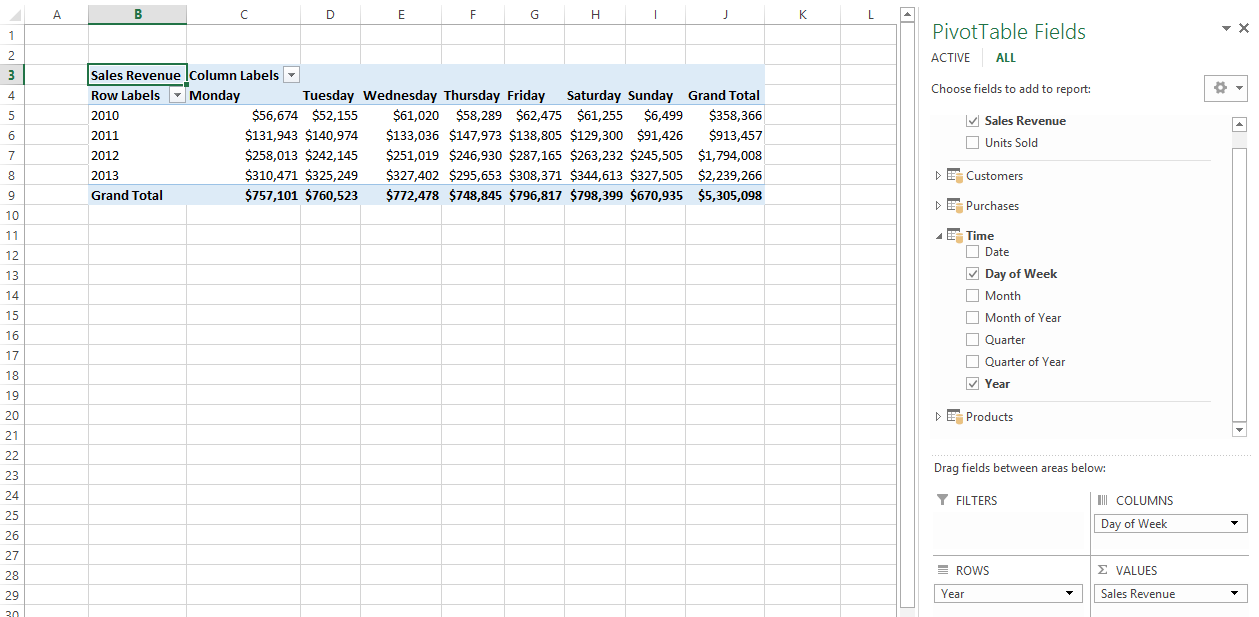
1. It's time to experiment by creating a different layout for the exact same data. Switch the **Year** column and the **Month of Year** column around so that the **Month of Year** column is in the **COLUMNS** section and the **Year** column is in the **ROWS** section. Notice how this changes the way the data is displayed.



1. Remove the **Month of Year** column from the **COLUMNS** section and replace it with the **Quarter of Year** column from the **Time** table to show yearly results by quarter instead of by month.



1. Remove the **Quarter of Year** column from the **COLUMNS** section and replace it with the **Day of Week** column from the Time table so you can see how sales have done on each particular day of the week on a yearly basis.

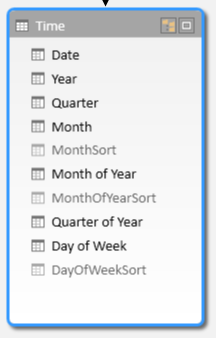


1. Save your work by saving your changes to **WingtipSalesModel.xlsx**.

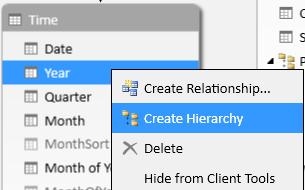
### Exercise 2: Creating a Calendar Drilldown Hierarchy in the Time Table

In this exercise you will create a hierarchy in the Time table to allow users to drill down into more specific periods of time. Then you will create a new PivotTable to test your work.

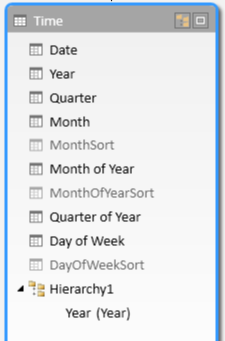
1. Return to the PowerPivot window and make sure you viewing the data model in Diagram View.
2. Inspect the current state of the **Time** table in **Diagram View**.



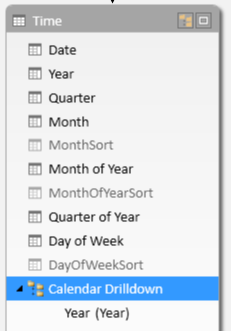
1. Right-click on the **Year** column and then click the **Create Hierarchy** command.



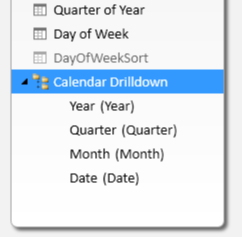
1. When you execute the **Create Hierarchy** command, you should see that the new hierarchy is created at the bottom of the **Time** table and given a default name of **Hierarchy1**.



1. Rename the new hierarchy from **Hierarchy1** to **Calendar Drilldown**.

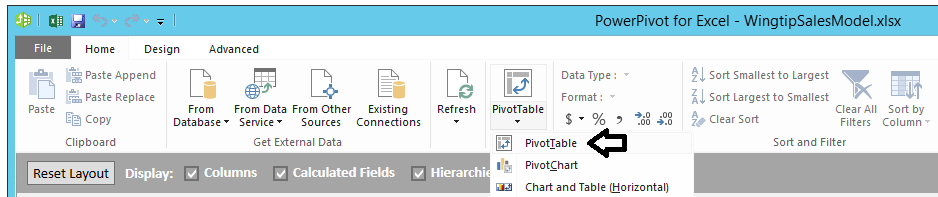


1. Add the columns named **Quarter**, **Month** and **Date** into the hierarchy.

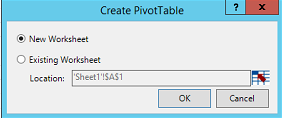


#### Create a PivotTable to uses the Calendar Drilldown Hierarchy

1. Create a new PivotTable that makes use of the **Time** table hierarchy you just created.
   1. From the **Home** tab in the ribbon of the **POWERPIVOT** window, drop down the **PivotTable** menu control and click **PivotTable** to create a new pivot table in in the workbook **WingtipSalesModel.xslx**.



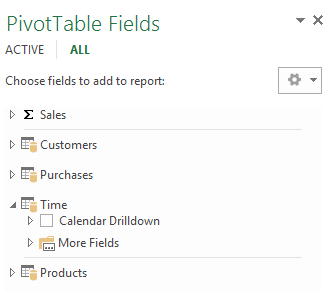
* 1. When the **Create PivotTable** dialog appears, select **New Worksheet** and click the **OK** button.



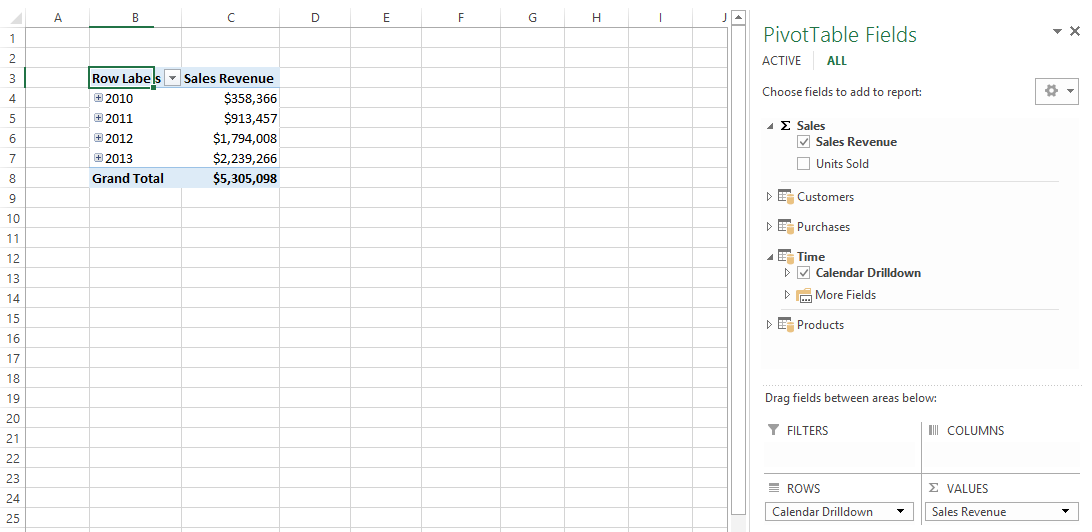
* 1. Once the new worksheet with the new PivotTable is added to the workbook, rename the worksheet **Sales Revenue Drilldown**.



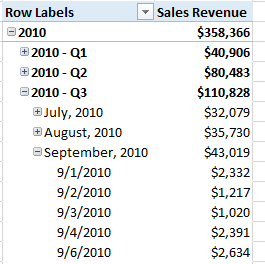
1. When you begin working on the new PivotTable, you should notice that the **Calendar Drilldown** hierarchy can now be seen inside the **PivotTable Fields** section in the **Time** table.



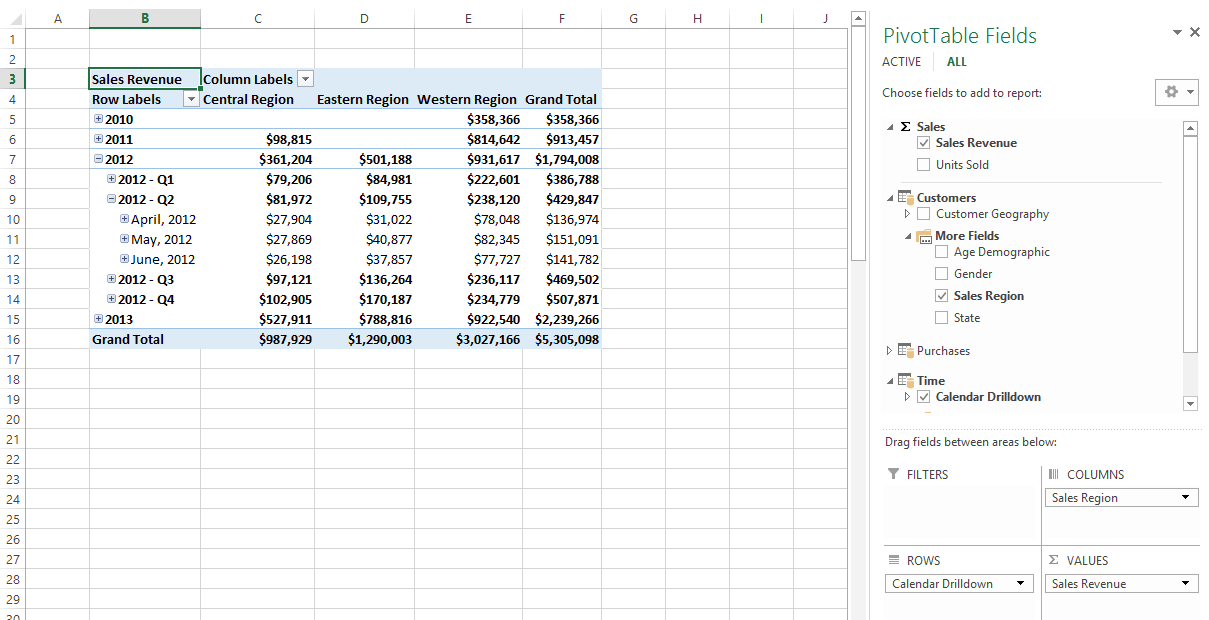
1. Create new pivot table by adding the **Sales Revenue** calculated fields from the **Sales** table to the **VALUES** section and the **Calendar Drilldown** hierarchy of the **Time** table to the **ROWS** section as shown in the following screenshot.



1. Now experiment by drilling down into the pivot table to drill into sales revenue by quarter, month and day.



1. Modify the PivotTable by adding the **Sales Region** column from the **Customers** table into the **COLUMNS** section. Your PivotTable should look like the one shown in the following screenshot.



1. Save your work by saving your changes to **WingtipSalesModel.xlsx**.

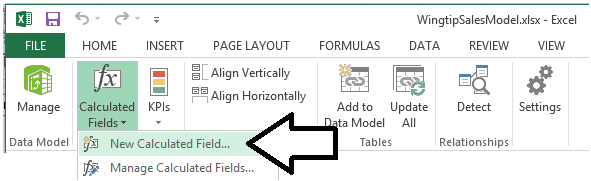
### Exercise 3: Creating Calculated Fields with Time Intelligence Function

In this exercise, you will leverage various DAX Time Intelligence functions to analyze sales revenue using quarter to date (QTD) totals year to date (YTD) totals. You will also create a DAX formula to calculate a running total of sales revenue through the entire 4 years of sales activity.

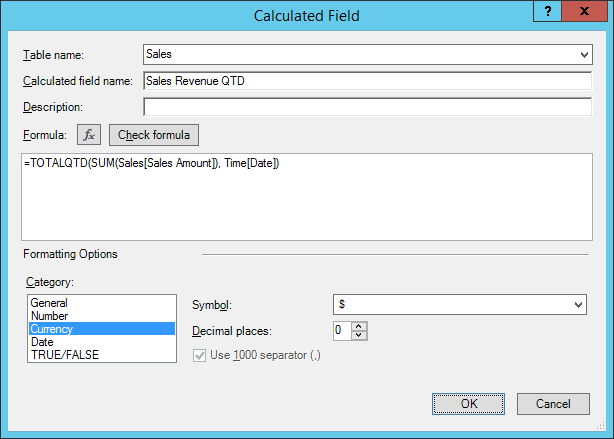
1. Navigate to the Excel application windows of **WingtipSalesModel.xlsx** if you are not already there.

#### Create Calculated Fields for Sales Revenue QTD and Sales Revenue YTD

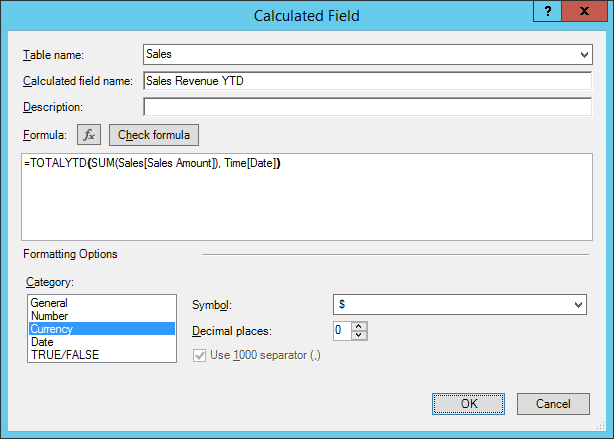
1. Create the **Sales** **Revenue QTD** calculated field.
   1. Drop down the Calculate Fields ribbon button menu and click **New Calculated Field…**.



* 1. Fill out the **Calculated Field** dialog with the following values.
     1. **Table name**: Sales
     2. **Calculated field name**: Sales Revenue QTD
     3. **Formula**: =TOTALQTD( SUM(Sales[Sales Amount]), Time[Date] )
     4. **Formatting Option Category**: Currency
     5. **Decimal places**: 0
  2. When the **Calculate Field** dialogs looks the following screenshot, click the **OK** button to create the calculated field.



1. Create the **Sales Revenue YTD** calculated field.
   1. Drop down the Calculate Fields ribbon button menu and click **New Calculated Field…**.
   2. Fill out the **Calculated Field** dialog with the following values.
      1. **Table name**: Sales
      2. **Calculated field name**: Sales Revenue YTD
      3. **Formula**: =TOTALYTD( SUM(Sales[Sales Amount]), Time[Date] )
      4. **Formatting Option Category**: Currency
      5. **Decimal places**: 0
   3. When the **Calculate Field** dialogs looks the following screenshot, click the **OK** button to create the calculated field.



#### Create the Sales Revenue Running Total calculated field using the CALCULATE function

1. Create the **Sales Revenue Running** calculated field.
   1. Drop down the Calculate Fields ribbon button menu and click **New Calculated Field…**.
   2. Fill out the **Calculated Field** dialog with the following values.
      1. **Table name**: Sales
      2. **Calculated field name**: Sales Revenue Running Total
      3. **Formatting Option Category**: Currency
      4. **Decimal places**: 0
   3. Add the following DAX expression for the calculate field formula.

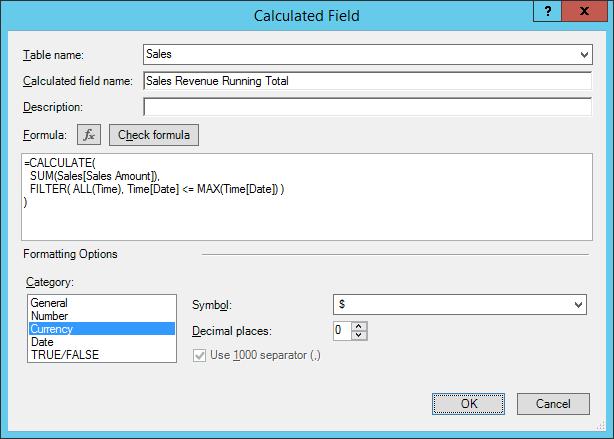
=CALCULATE(

SUM(Sales[Sales Amount]),

FILTER( ALL(Time), Time[Date] <= MAX(Time[Date]) )

)

* 1. When the **Calculate Field** dialogs looks the following screenshot, click the **OK** button to create the calculated field.

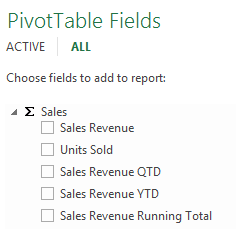


#### Create a PivotTable to use the new Calculated Fields which use Time Intelligence

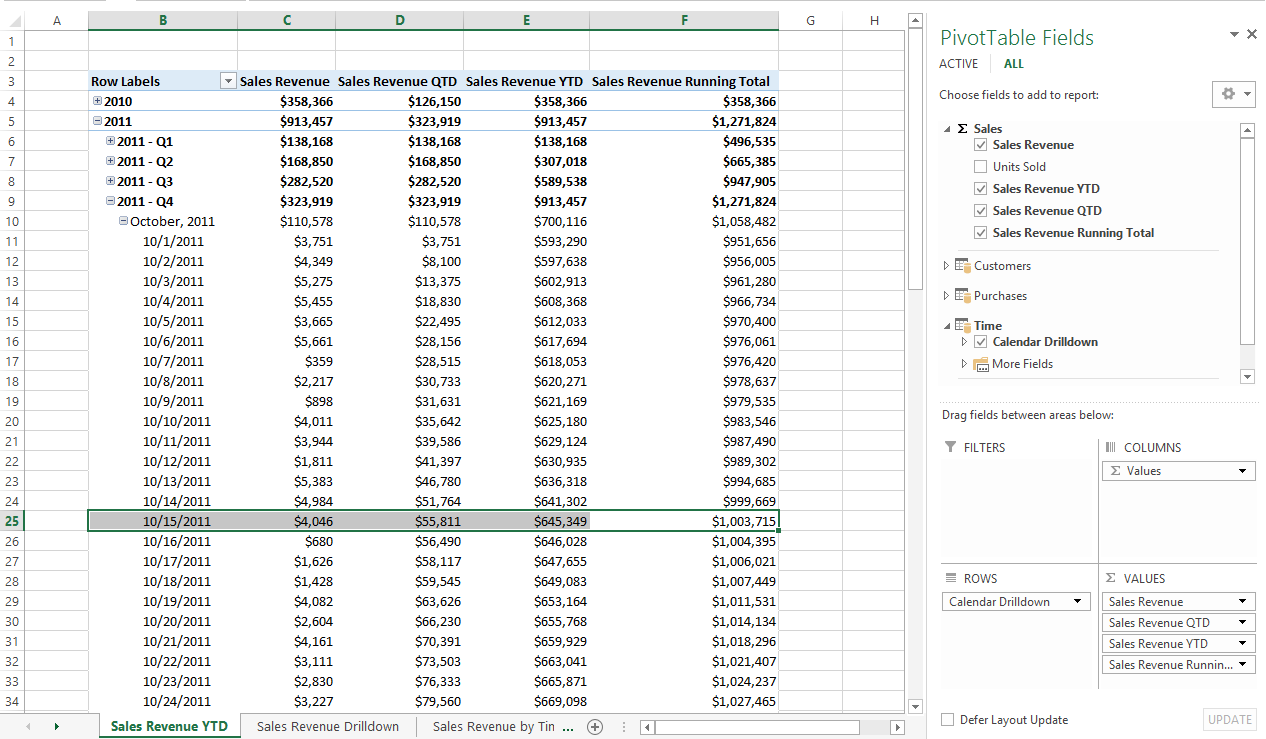
1. Create a new PivotTable that makes use of the three new calculated fields you just created.
   1. From the **Home** tab in the ribbon of the **POWERPIVOT** window, drop down the **PivotTable** menu control and click **PivotTable** to create a new pivot table in in the workbook **WingtipSalesModel.xslx**.
   2. When the **Create PivotTable** dialog appears, select **New Worksheet** and click the **OK** button.
   3. Once the new worksheet with the new PivotTable is added to the workbook, rename the worksheet **Sales Revenue Total**.



1. When you begin working on the new PivotTable, you should notice that the three calculated fields named Sales Revenue hierarchy can now be seen inside the **PivotTable Fields** section in the **Time** table.



1. Create the PivotTable by adding **Sales Revenue**, **Sales Revenue QTD**, **Sales Revenue YTD** and **Sales Revenue Running Total** in the **VALUES** section. Add the **Calendar Drilldown** hierarchy into the **ROWS** section so that your PivotTable appears as one in the following screenshot. Locate the day where Wingtip Toys first reached $1,000,000 in total sales revenue.



You have now learned how to use DAX Time Intelligence functions together with a time dimension table. Now you will move on to the final exercise where you will create a Key Performance indicator(KPI) to monitor sales growth.

### Exercise 4: Creating a KPI to Monitor Sales Growth

In this exercise you will create a new calculated field to calculate the growth of sales revenue on a month-by-month basis. After that you will convert this calculated field into a KPI which will be used to monitor sales growth and provide visual indications as to how each month has done when compared to the previous month.

#### Create a Calculated Field to Calculate Sales Growth from the Previous Month

1. Create the **Sales Growth from PM** calculated field.
   1. Drop down the **Calculate Fields** ribbon button menu and click **New Calculated Field…**.
   2. Fill out the **Calculated Field** dialog with the following values.
      1. **Table name**: Sales
      2. **Calculated field name**: Sales Growth from PM
      3. **Formatting Option Category**: Currency
      4. **Decimal places**: 0
   3. Add the following DAX expression for the calculate field formula or copy it from **SalesGrowthForPMDAX.txt** in the lab folder.

=IF(

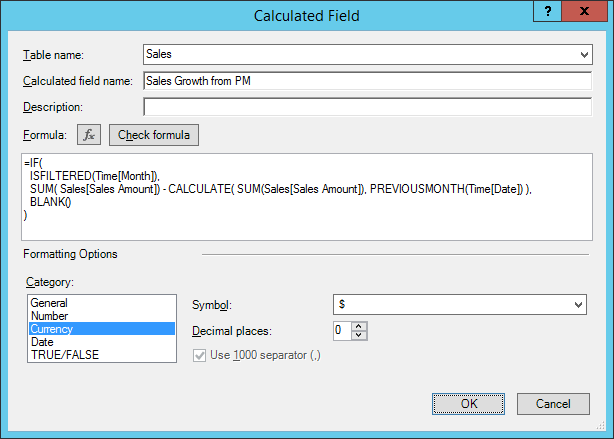
ISFILTERED(Time[Month]),

SUM(Sales[Sales Amount]) - CALCULATE( SUM(Sales[Sales Amount]), PREVIOUSMONTH(Time[Date]) ),

BLANK()

)

* 1. When the **Calculate Field** dialogs looks the following screenshot, click the **OK** button to create the calculated field.



#### Create a Calculated Field to Calculate the Percentage of Sales Growth from the Previous Month

1. Create the **Sales Growth** calculated field.
   1. Drop down the **Calculate Fields** ribbon button menu and click **New Calculated Field…**.
   2. Fill out the **Calculated Field** dialog with the following values.
      1. **Table name**: Sales
      2. **Calculated field name**: Sales Growth
      3. **Formatting Option Category**: Number
      4. **Format**:Percent
   3. Type in the following DAX expression for the calculate field formula or copy it from **SalesGrowthDAX.txt** in the lab folder.

=IF(

( ISFILTERED(Time[Month]) && ISFILTERED(Time[Date])=FALSE() ),

DIVIDE(

SUM(Sales[Sales Amount]) -

CALCULATE(

SUM(Sales[Sales Amount]),

PREVIOUSMONTH(Time[Date])

),

CALCULATE(

SUM(Sales[Sales Amount]),

PREVIOUSMONTH(Time[Date])

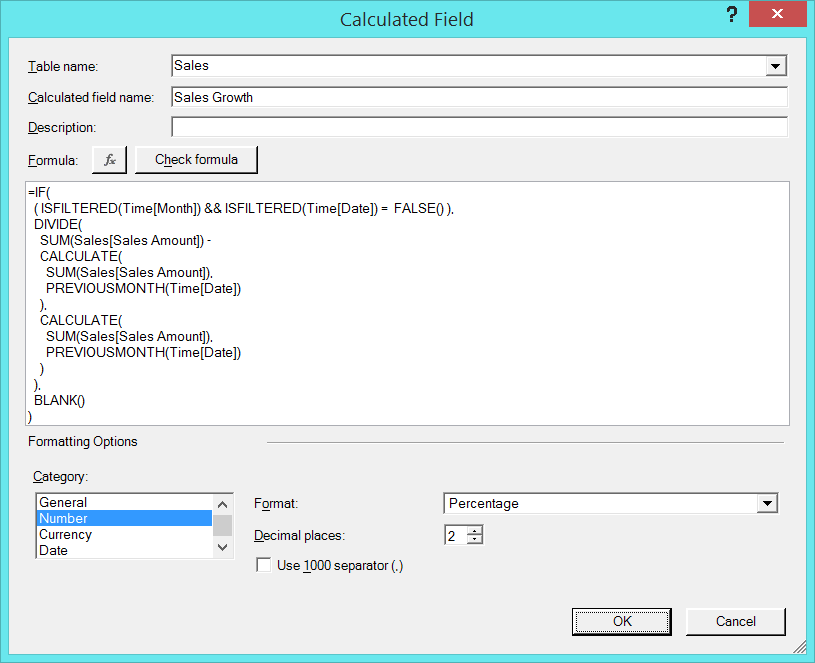
)

),

BLANK()

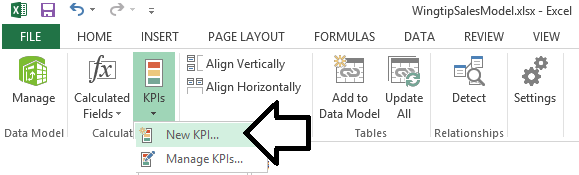
)

* 1. When the **Calculate Field** dialogs looks the following screenshot, click the **OK** button to create the calculated field.

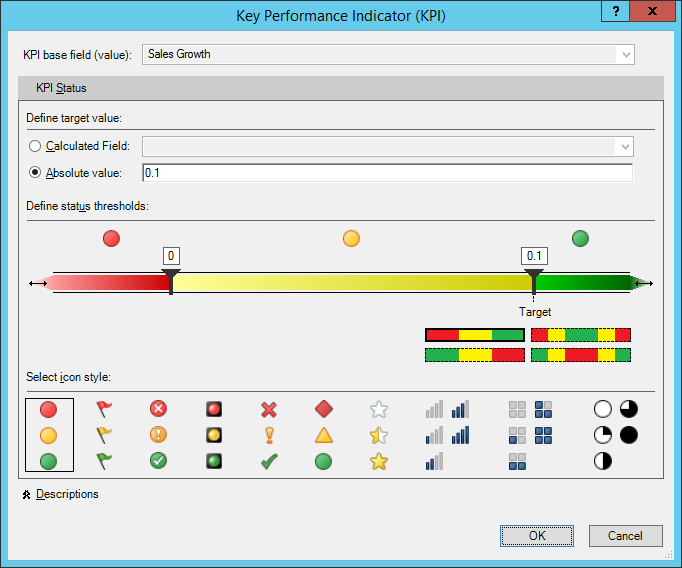


#### Convert the Sales Growth Calculated Field into a KPI

1. Create a new KPI based on the **Sales** **Growth** calculated field.
   1. Drop down the Calculate Fields ribbon button menu and click **New KPI…**.



* 1. Fill out the **Key Performance Indicator (KPI)** dialog with the following values
     1. **KPI base field (value)**: Sales Growth
     2. **Define target value**: Absolute value of 0.1
     3. **Lower threshold**: 0
     4. **Upper Threshold**: 0.1
  2. When you **Key Performance Indicator (KPI)** looks like the following screenshot, click the **OK** button to create the new KPI.



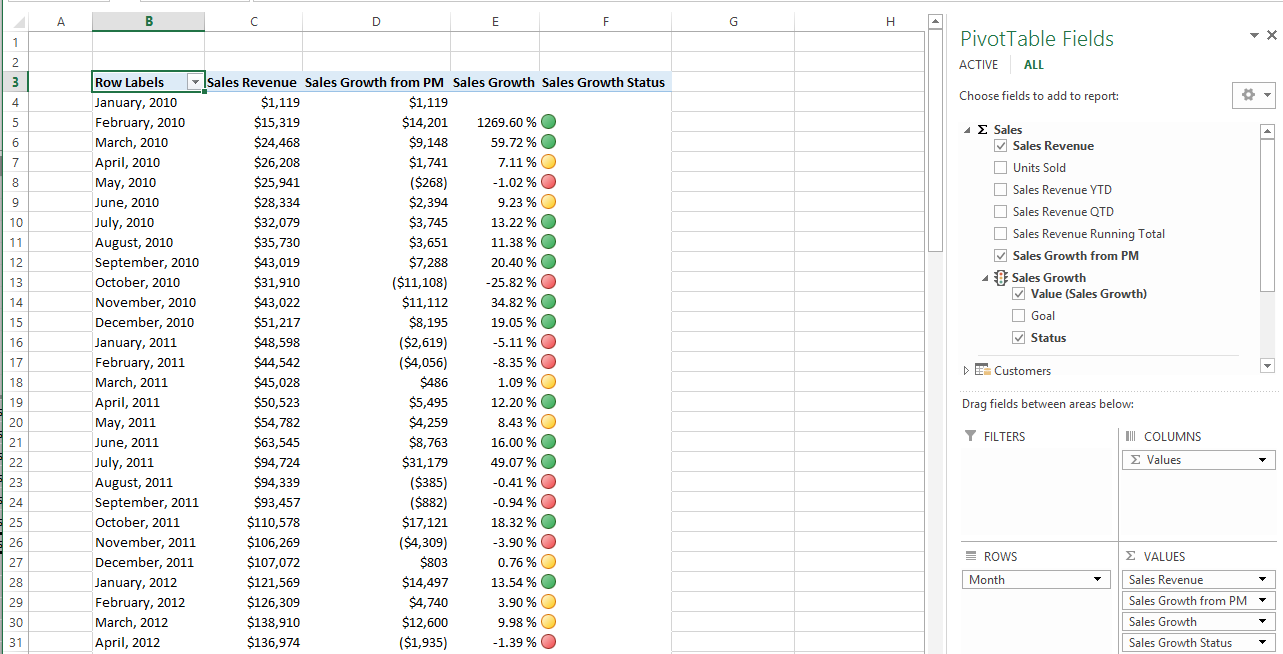
Now that you have created a new KPI, it's time to use it within a PivotTable

#### Create a PivotTable to use the Sales Growth KPI

1. Create a new PivotTable that makes use of the three new calculated fields you just created.
   1. From the **Home** tab in the ribbon of the **POWERPIVOT** window, drop down the **PivotTable** menu control and click **PivotTable** to create a new pivot table in in the workbook **WingtipSalesModel.xslx**.
   2. When the **Create PivotTable** dialog appears, select **New Worksheet** and click the **OK** button.
   3. Once the new worksheet with the new PivotTable is added to the workbook, rename the worksheet **Sales Growth**.



1. Create the new PivotTable by adding the **Month** column of the **Time** table into the **ROWS** section and then add the calculate fields **Sales Revenue** and **Sales Growth from PM** into the **VALUES** section. Complete our work by adding the **Value** and **Status** fields of the **Sales Growth** KPI into the **VALUES** as shown in the following screenshot.



You are now done with this lab. You accomplished quite a bit in learning about using time dimension tables, DAX Time Intelligence functions and KPIs in an Excel data model.