## Extracting and Transforming Data using Power Query

**Lab Time**: 40 minutes

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

**Lab Overview**: In this lab you will work with Power Query to extract and transform data that is loaded into an Excel workbook. You will begin by extracting data from a text-based CSV file and creating a simple set of Power Query transform steps. After that, you will use Power Query to extract data from a SharePoint list and from tables in an Access database that have pre-existing relationships.

### Exercise 1: Install the Power Query Add-in for Excel (if it's not already installed)

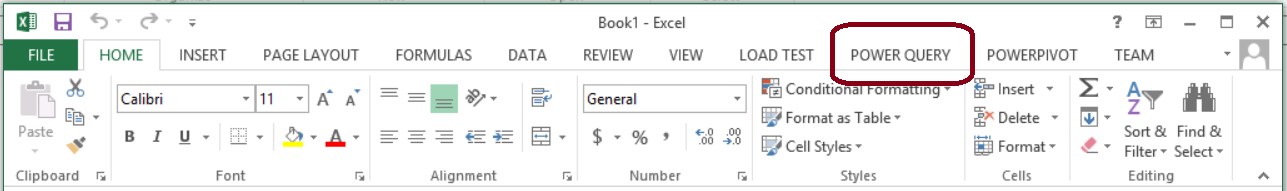
Before you can begin to work with Power Query, you must make sure it has been installed and activated with Microsoft Excel 2013.

#### Log in to Student Environment

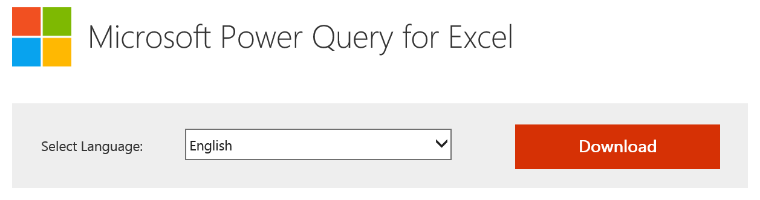
1. Login to the Student VM using the login **WINGTIP\Administrator** and the appropriate password.
   1. If you’re using a local VM provided by the hosting training company, the password will be **Password1**.
   2. If your student VM is hosted by CloudShare, the password for the **WINGTIP\Administrator**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

#### Enable Excel Add-Ins

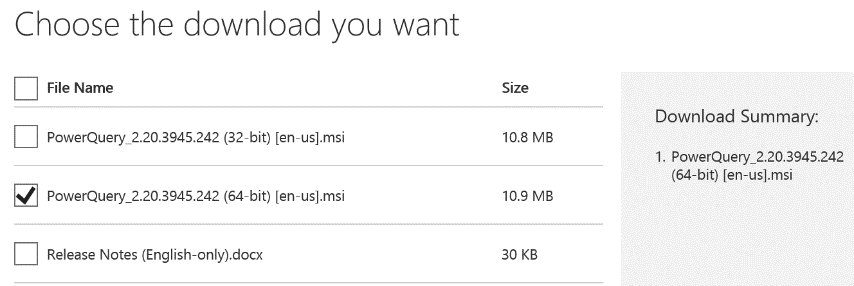
1. Launch Microsoft Excel 2013.
   1. Press the **Windows** key to navigate to the Windows Start page.
   2. Launch Microsoft Excel 2013.
   3. Create a new Blank workbook.
   4. Check to see if there is a **Power Query** tab in the ribbon in the Excel application window. If the **Power Query** tab exists, it means that the Power Query add-in is installed and activated which then means you can skip the remaining steps of this exercise and move ahead to Exercise 2. If there is no Power Query tab, then you must complete the steps in this exercise to install and activate the Power Query add-in.



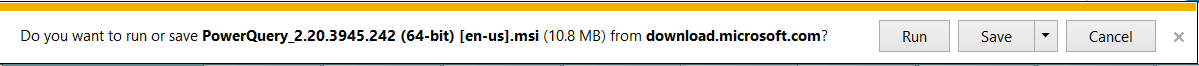
1. Close the Excel application window if it is still open.
2. Download and Install the Power Query add-in.
   1. Open Internet Explorer and navigate to <http://www.microsoft.com/en-us/download/details.aspx?id=39379>
   2. On the **Microsoft Power Query for Excel** download page, click the **Download** button.



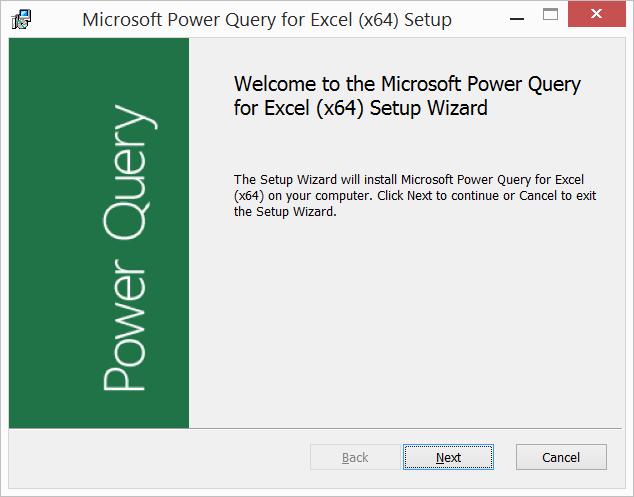
* 1. On the **Choose the download you want** page, select the 64-bit version of Power Shell as shown above and click **Next**.



* 1. When prompted by Internet Explorer to run or save the Power Query installation file, click the **Run** button to begin the download and installation of the Power Query add-in.

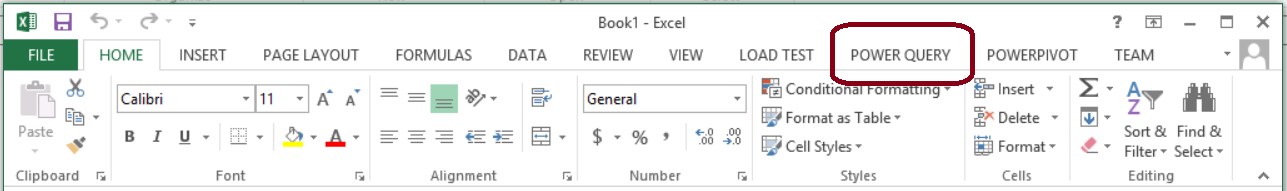


* 1. When the Microsoft Power Query for Excel Setup begins, it prompts you with a greeting dialog as shown in the following screenshot. Click **Next** to continue the installation.



* 1. When prompted by the Power Query setup program, agree to Microsoft's terms of use and follow the steps to complete the installation.
  2. After the installation has completed, you should be prompted by Power Query Setup Wizard dialog indicating that the installation of Power Query was successful.

1. Launch Microsoft Excel 2013 and verify that the Power Query add-in has been installed and activated.
   1. Launch Microsoft Excel 2013 and create a new Blank workbook.
   2. Check to verify that there is a **Power Query** tab in the ribbon in the Excel application window.

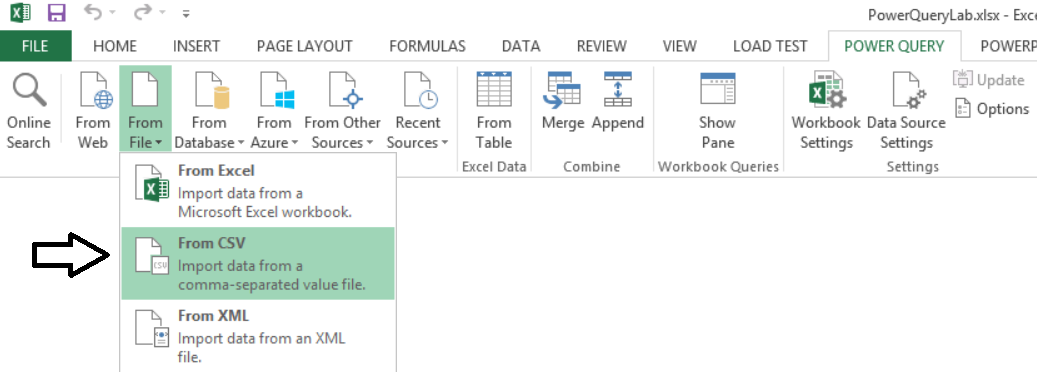


* 1. Once you have verified that the Power Query add-in has been installed and activated, you can move on to the next exercise where you will begin to use Power Query to extract and transform data.

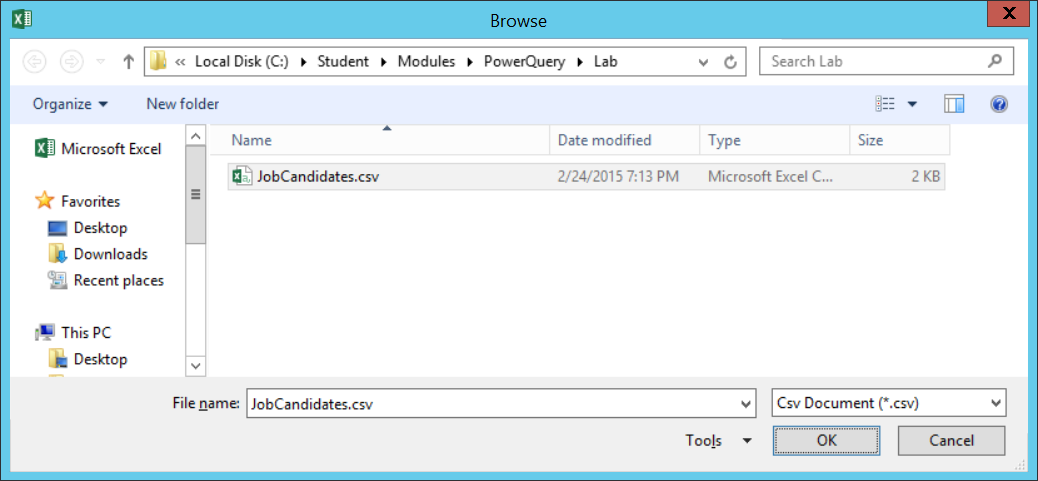
### Exercise 2: Creating a Simple Query using the Query Editor

In this exercise you will use Power Query to create a simple query to extract data from a CSV file. This will give you an opportunity to begin working with the Query Editor window and to perform common tasks with Power Query such as adding and configuring steps.

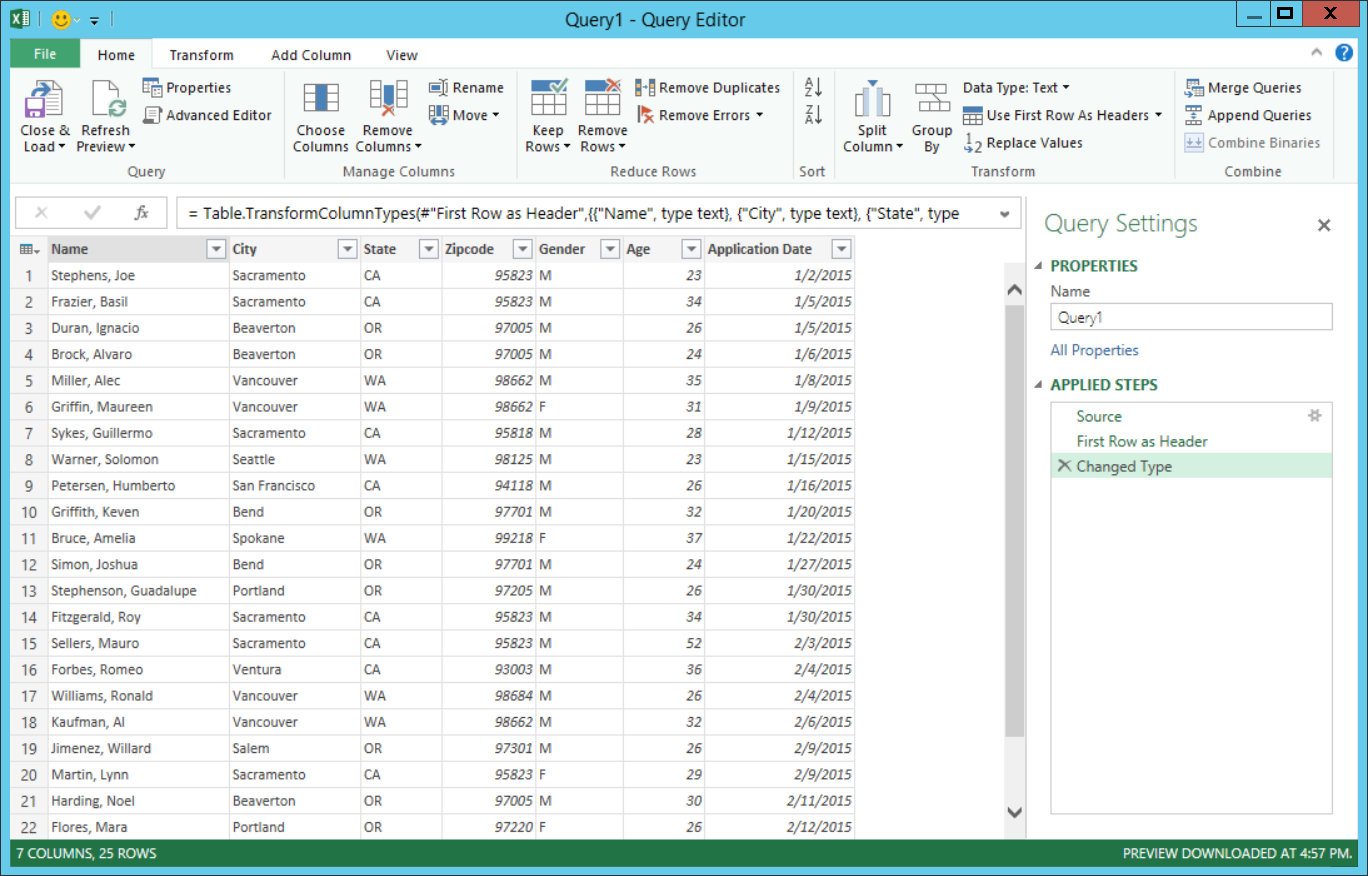
1. Create a new Excel workbook file.
   1. Launch Microsoft Excel 2013 if it is not already running.
   2. Close any existing workbook that is currently open.
   3. Create a new Blank workbook.
   4. Save the new workbook file as **PowerQueryLab.xlsx** in the folder **C:\Student\Modules\PowerQuery\Lab**.
2. Create a new query from data inside a CSV file named **JobCandidates.csv**.
   1. In the Excel application window, click on the **Power Query** tab.
   2. Inside the **Get External Data** ribbon group, drop down the **From File** menu and select the **From CSV** command.



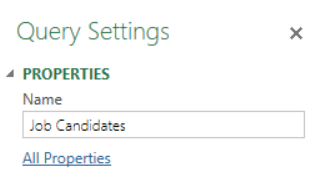
* 1. In the **Browse** dialog, navigate to the folder at **C:\Student\Modules\PowerQuery\Lab\** and select the file named **JobCandidates.csv** and click the **OK** button.



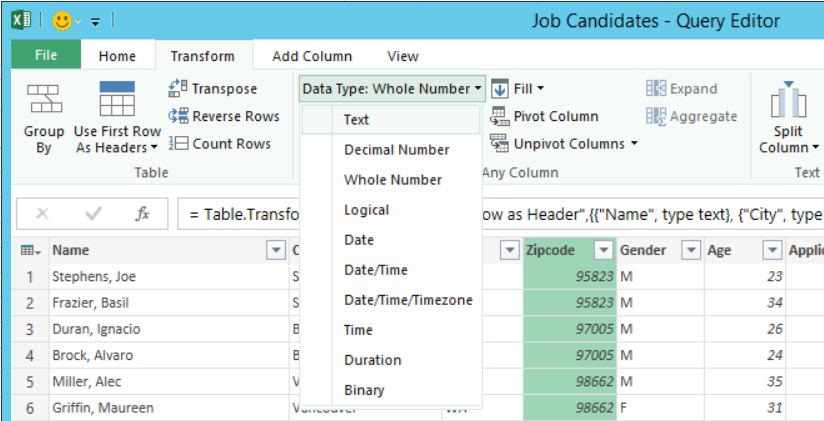
* 1. When you click **OK**, Power Query should create a new query which is opened in the Query Editor window. If you look at the **Query Properties** pane on the right, you should be able to see that the new query has a given a generic name such as **Query1**.



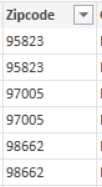
* 1. In the **Query Settings** pane of the Query Editor window, update the query **Name** property to **Job Candidates**.



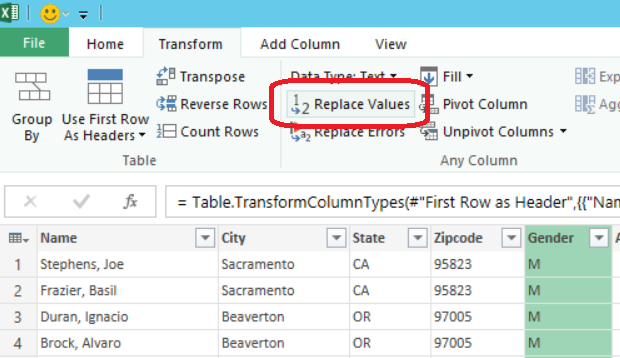
1. Modify column named **Zipcode** to be a Text column.
   1. Inspect the column named **Zipcode** in the Query Editor window. As you can see, the data in this column is aligned to the right which indicates this column is currently recognized as a numeric column instead of a text-based column. In this step you will explicitly configure the column to be based on the **Text** column type.
   2. Select the **Zipcode** column by clicking the column header. Next, drop down the **Data Type** menu in the ribbon and set the column type to **Text** as shown in the following screenshot.



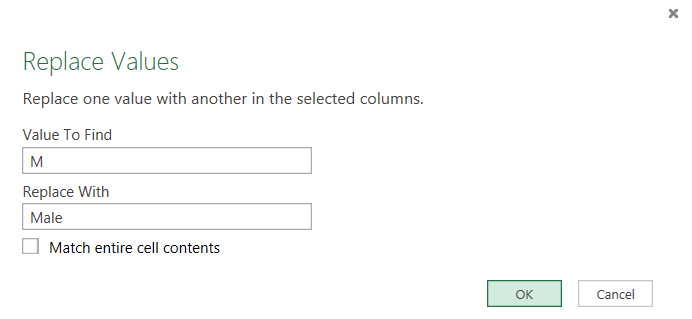
* 1. After you have modified the **Zipcode** column to be a **Text** column, you should see that the data in the column is now aligned to the left instead of the right.



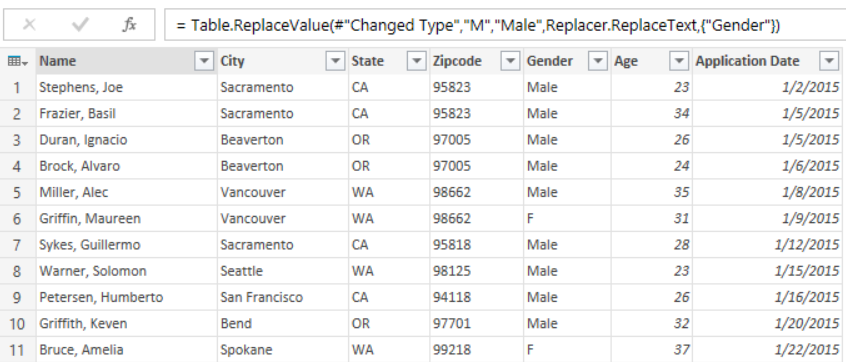
1. Replace Gender values of "M" and "F" with "Male" and "Female" to make them more readable.
   1. Select the **Gender** column by clicking its column header.
   2. From the **Transform** tab of the ribbon, click the **Replace Values** button.



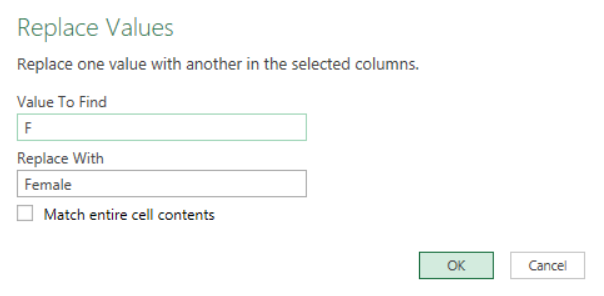
* 1. When prompted with the **Replace Values** dialog, enter a value of **M** in the **Value To Find** textbox and a value of **Male** in the **Replace With** textbox as shown on the following screenshot. Click the OK button to save this step.



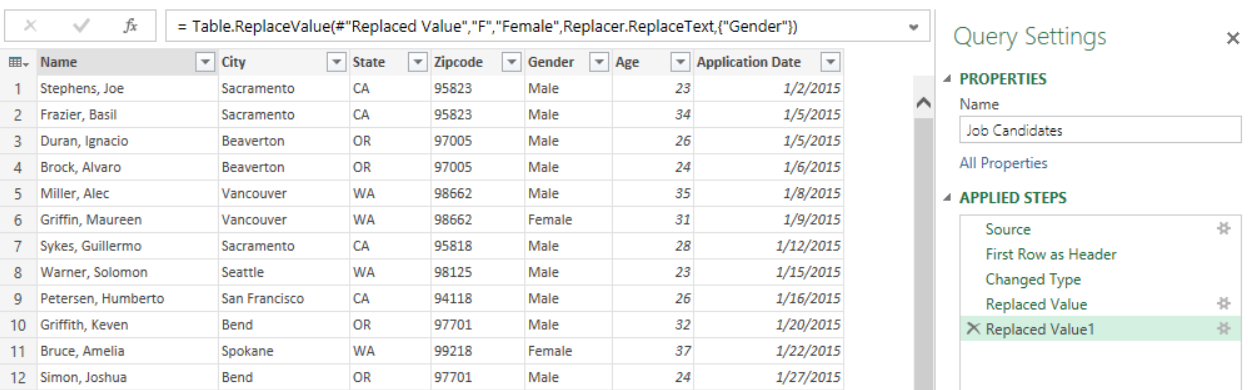
* 1. Now when you inspect the **Gender** column values in the Query Editor window, you should see all the values of **M** have been replaced with **Male**. Now it's time to switch of the **F** values with **Female**.



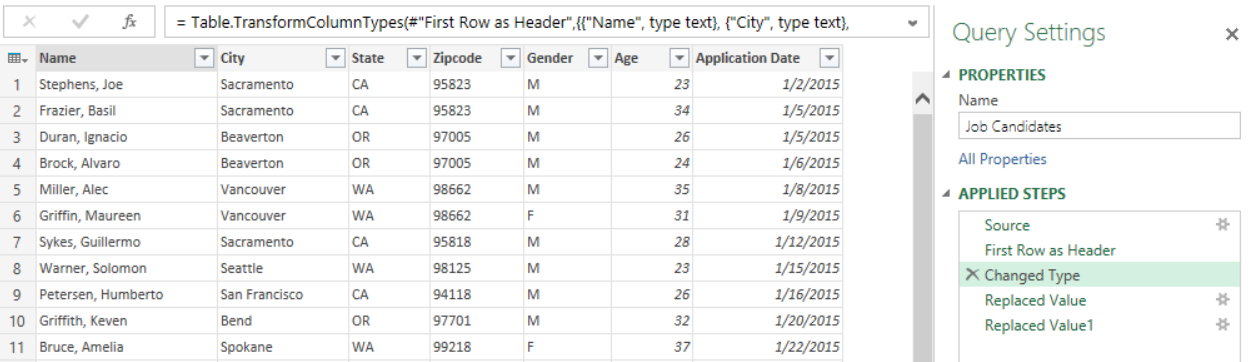
* 1. Select the **Gender** column by clicking its column header.
  2. From the **Transform** tab of the ribbon, click the **Replace Values** button.
  3. When prompted with the **Replace Values** dialog, enter a value of **F** in the **Value To Find** textbox and a value of **Female** in the **Replace With** textbox as shown on the following screenshot. Click the **OK** button to save this step.



* 1. At this point, all value in the Gender column should be either **Male** or **Female**.



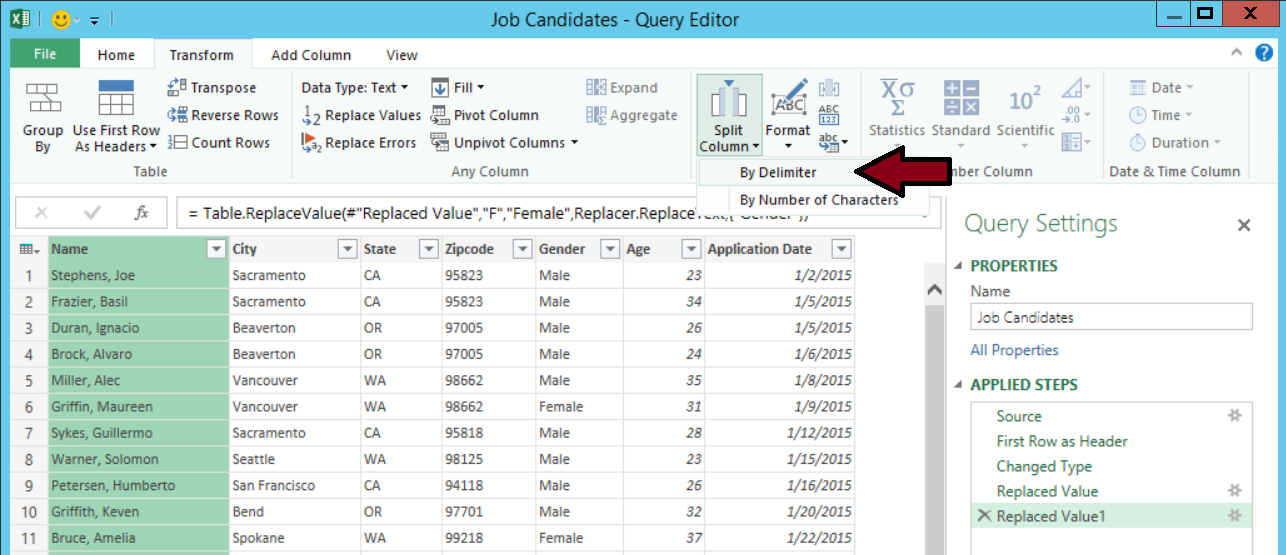
1. Now that you have created several steps in the new query, it is time to see how you can replay these steps by clicking on specific steps in the **APPLIED STEPS** view in the **Query Settings** pane.
   1. In the **APPLIED STEPS** view, click on the first step name **Source**. Note that the transforms performed in later steps are removed from the Query Results pane of the Query Window.
   2. Next click on the **First Row as Header** step and then click the **Changed Type** step to see how it effects what is displayed in the Query Results pane.



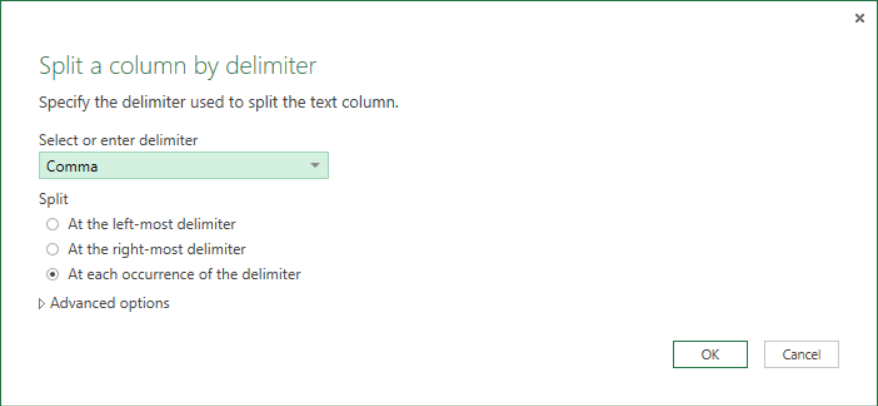
* 1. Finally click the two remaining steps until you reach the final step.

In most cases, you should select the final step in the **APPLIED STEPS** pane before performing a new transformation. This will cause the new transformation to be added at the very end. If you select a step in the middle and then perform a transformation, the new transformation will be added as a step in the middle of existing steps. While inserting new steps into the middle of a set of existing steps is sometime required, you must be careful because if can cause errors in later steps if you do something such as change the name of a column.

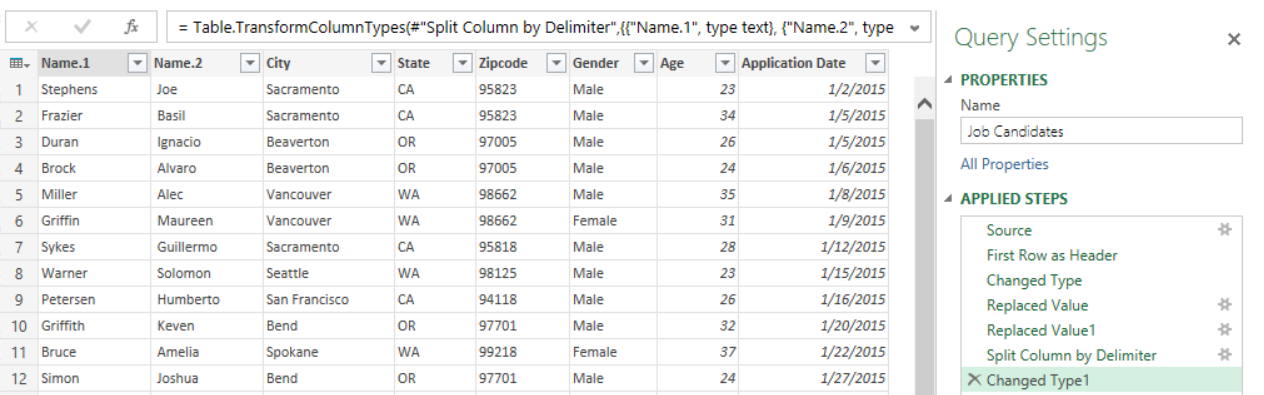
1. Split the Name column out in separate columns for First Name and Last Name.
   1. Select the **Name** column by clicking on its column header.
   2. On the **Transform** tab of the ribbon in the **Query Editor** window, look in the Transform group and locate the **Split Column** dropdown menu. Drop down the **Split Column** menu and select **By Delimiter**.



* 1. On the **Split a column by delimiter** dialog, select **Comma** in the **Select or enter delimiter** dropdown as shown in the following screenshot and click **OK** to save the new step.



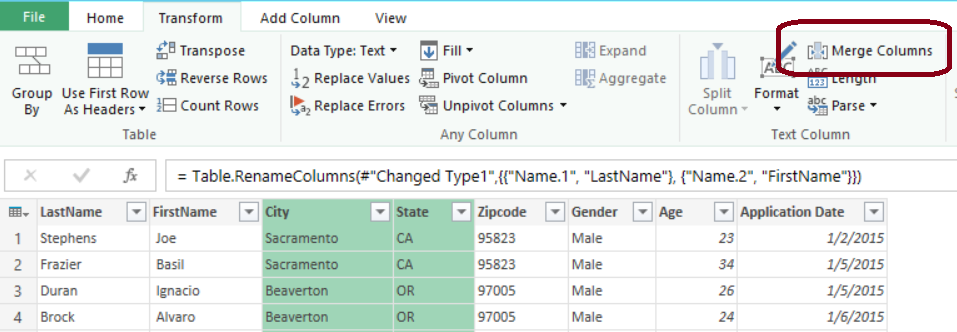
* 1. At this point, you should see that the **Name** column has been broken out into two columns named **Name.1** and **Name.2**.



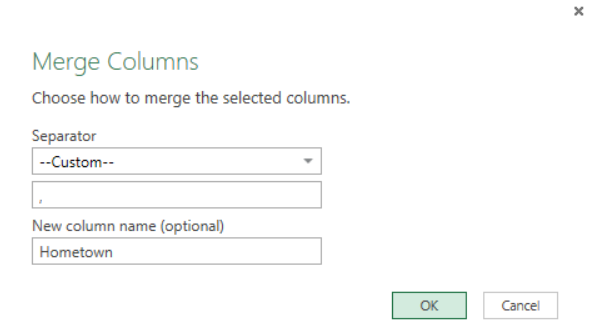
* 1. Rename the **Name.1** column to **Last Name** and rename the **Name.2** column to **First Name** as shown in the following screenshot.



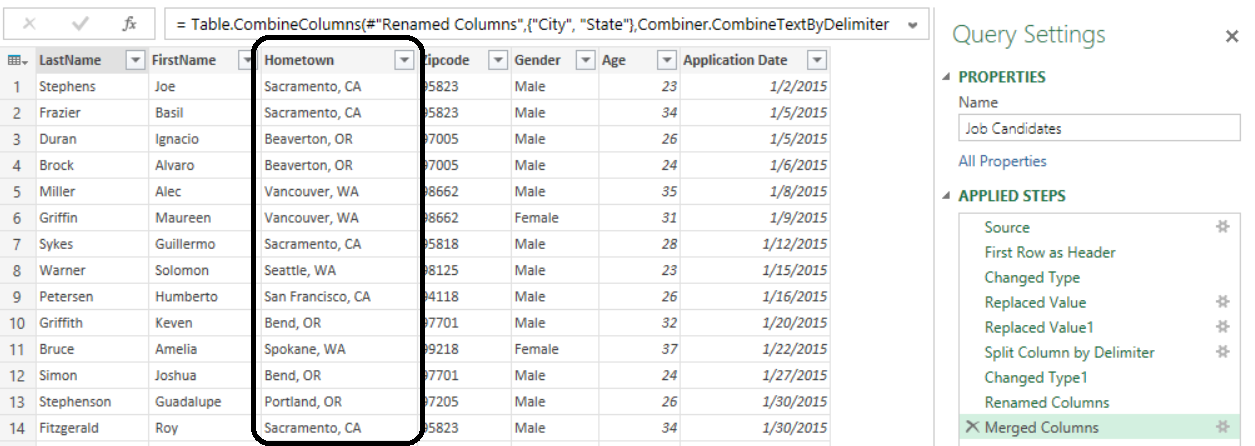
1. Now that you have seen how to split a column into two columns, now it's time to learn how to do the opposite and merge two columns into one. In this step you will merge the **City** column and the **State** column into a single column named **Hometown**.
   1. Select the **City** column by clicking on its column header.
   2. With the **City** column selected, hold down the **Shift** key on the keyboard and then use the mouse to additionally select the **State** column by clicking its column header. When both columns are selected as shown in the following screenshot, click the **Merge Columns** button which is located in the **Transform** tab in the ribbon inside the **Text Columns** group.



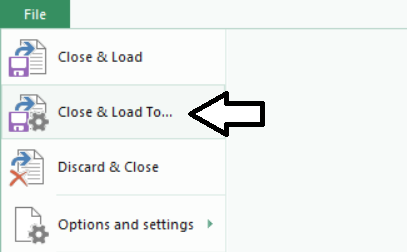
* 1. When the Marge Columns dialog appears, select **--Custom--** as the separator type and then type a comma (",") followed by a black space (" ") in the textbox below. In the **New column name** textbox, enter a column name of **Hometown**. Click **OK** when you are done to create the new step.



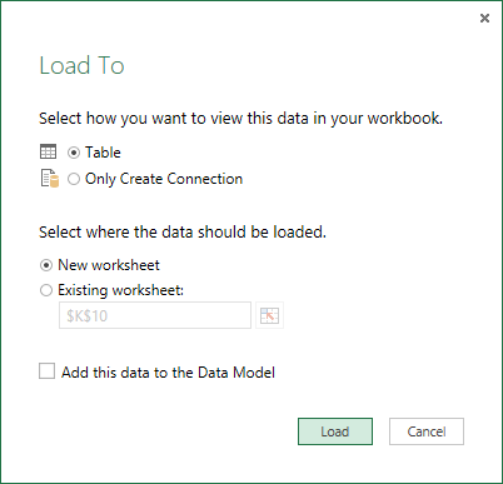
* 1. You should now see that the two columns named **City** and **State** have been removed from the query results window and replaced with a new column named **Hometown**.



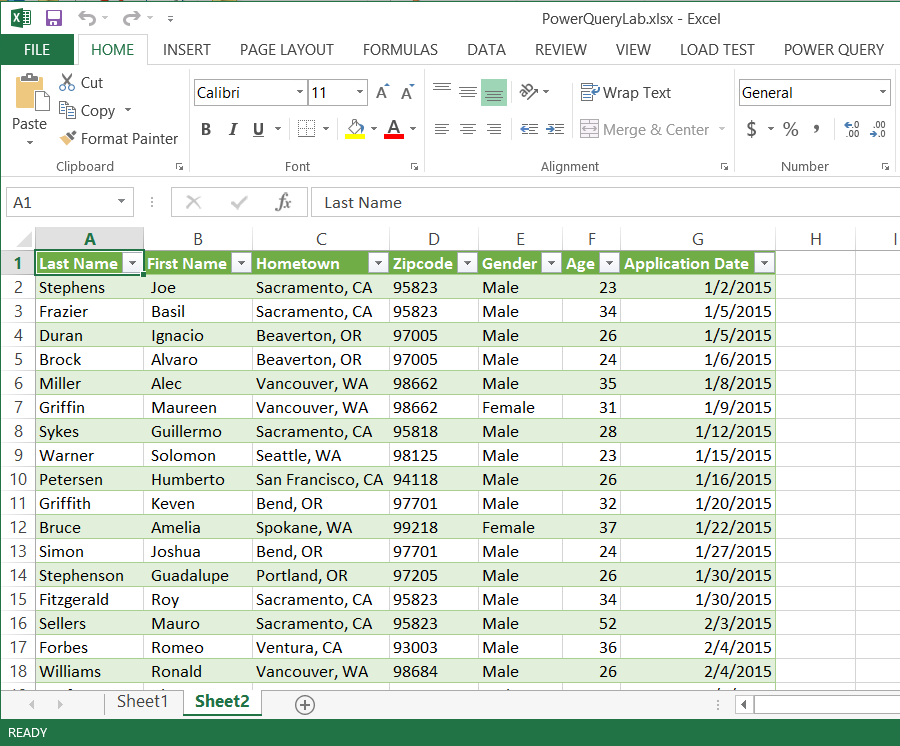
1. In this step you will save the query and configure its **Load To** settings.
   1. Drop down the **File** menu in the upper left-hand corner of the Query Editor window and click the **Close and Load To…**.



* 1. You should observe that the default settings are configured to create a new worksheet and load the query results into a new table created in the new worksheet as shown in the following screenshot. Click the **Load** button to run the query and load the query output into a new worksheet.



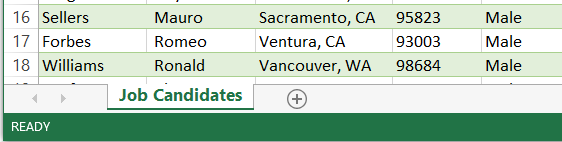
* 1. When you click the **Load** button, Power Query responds by adding a new worksheet to the current workbook so it has a place to load the query output into a new table in the new worksheet. Note that the new worksheet is given a generic name such as **Sheet2**.



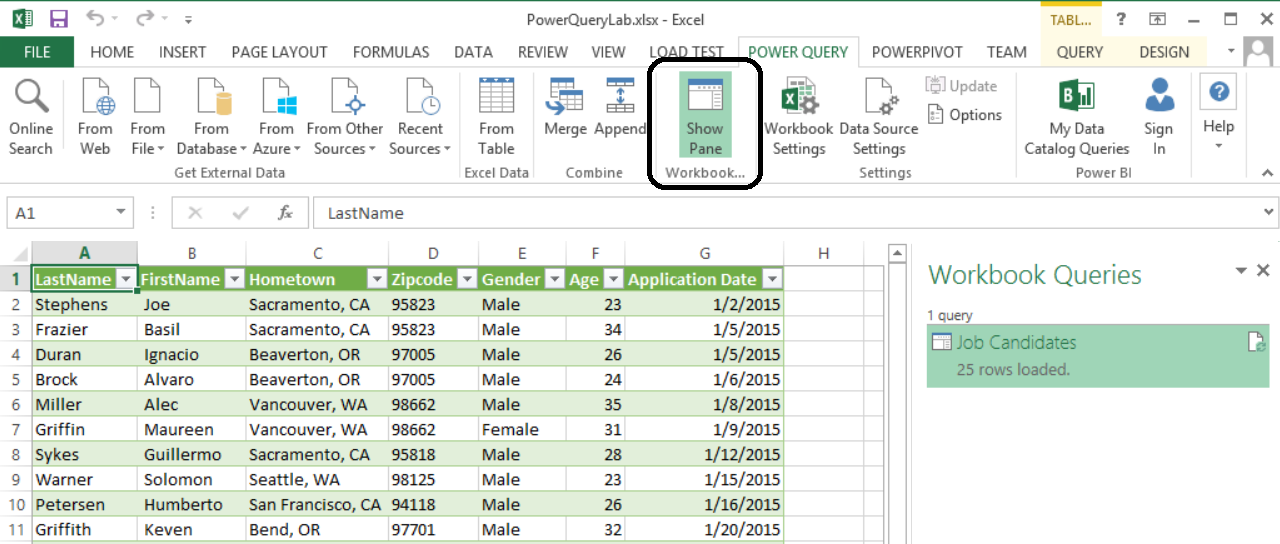
* 1. Rename the new worksheet to **Job Candidates**. You can accomplish this this by right-clicking the worksheet tab and selecting the **Rename** command.



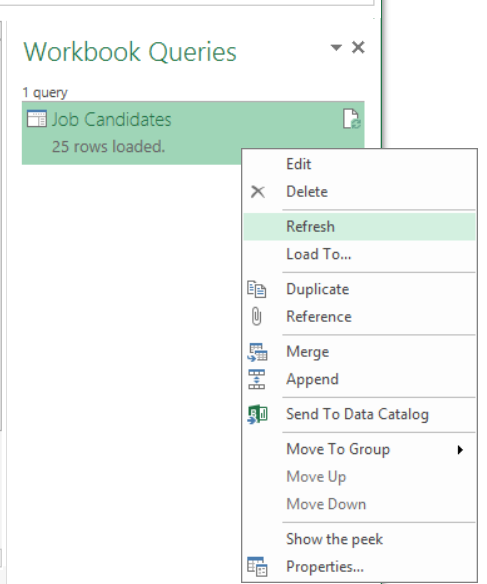
* 1. Delete the worksheet name **Sheet1** so that the **Job Candidates** worksheet is the only worksheet in the current workbook. You can accomplish this by right-clicking on the Sheet1 worksheet tab and selecting the **Delete** command.



1. Now that you have created a query, should be able to see it in the **Workbook Queries** pane in the task pane. In the Excel application window, navigate to the **Power Query** tab in the ribbon and click the **Show Pane** button to display the Workbook Queries pane as shown in the following screenshot. You should be able to see the Job Candidates query that you created in the previous steps of this exercise.



* 1. Right-click in the **Job Candidates** query in the **Workbook Queries** view. Take a moment and review the menu commands that are available to run on a query you have created with Power Query. Click on **Refresh** to rerun the query and reload the new results into the worksheet that was created earlier when you first loaded the **Job Candidates** query.

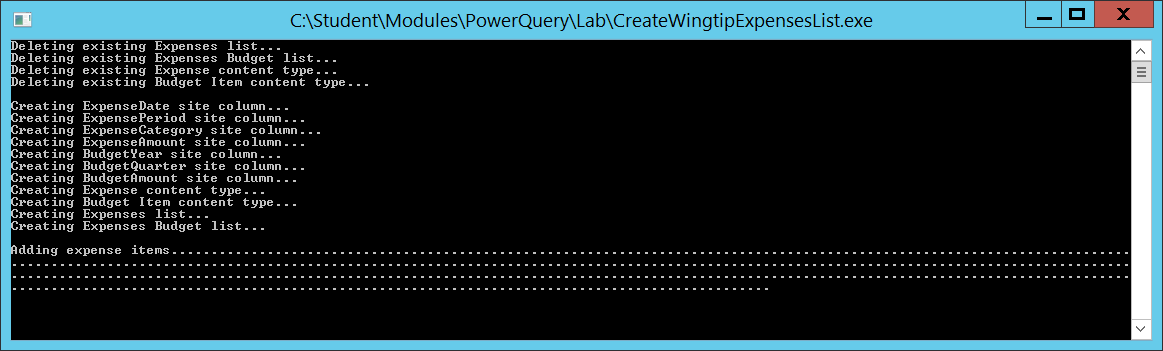


1. You have now reached the end of this exercise. Save your work by saving all changes to the Excel workbook file named **PowerQueryLab.xlsx**.

### Exercise 3: Using Power Query to Extract Data from a SharePoint List

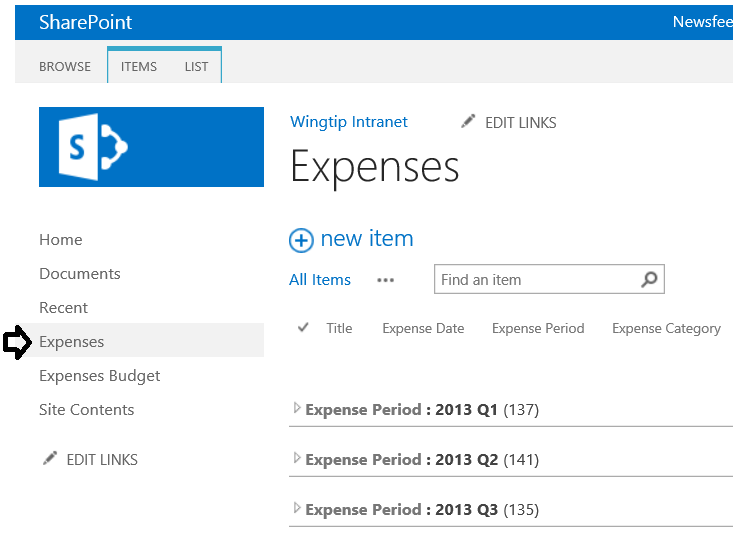
In this exercise you will learn how to extract data from a SharePoint list and import it into an Excel workbook.

1. Navigate to the SharePoint site at <http://intranet.wingtip.com>.
2. Run a utility program to create an **Expenses** list in the SharePoint site.
   1. Using Windows Explorer, navigate to the folder for this lab at **C:\Student\Modules\PowerQuery\Lab**.
   2. Locate and double click on the file named **CreateWingtipExpensesList.exe**. When you double click this file, it will launch a utility program which connects to the SharePoint site at <http://intranet.wingtip.com> and creates two lists. The first list is the **Expenses** list and the second list is the **Expenses Budget** list. It will take a minute or two for the utility program to complete its work in creating the lists and populating them with items. You should be able to see its progress in a Console window.

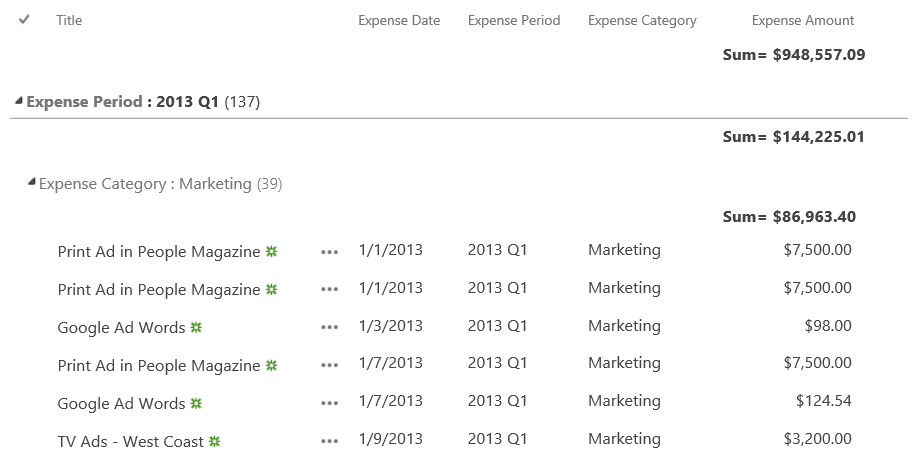


* 1. Wait until the utility program completes it work and then move on to the next step.

1. Ensure that the Expenses list has been created in the SharePoint site.
   1. Return to the browser and refresh the web page for the site at <http://intranet.wingtip.com>.
   2. You should be able to see a navigation link to the **Expenses** list in the **Quick Launch** menu. Click on the **Expenses** kink to navigate to the default view of the **Expenses** list.
   3. You should be able to see that the default view of the **Expenses** list has a collapsed top-level group for each financial quarter.

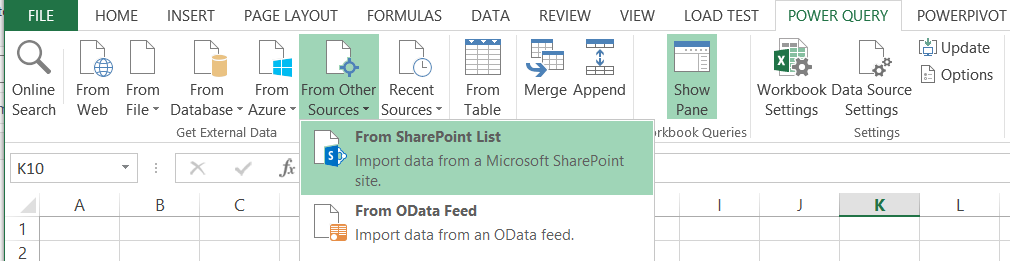


* 1. Expand the top group for the first quarter and then expand a category group to drill down to the level of expense items.

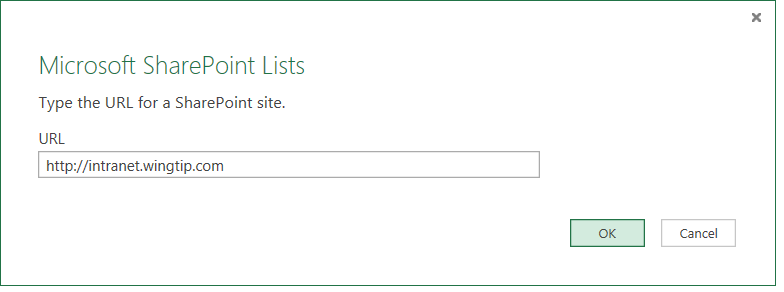


* 1. You have now verified that the **Expenses** list exists in the SharePoint site. In the next step you will use Power Query to extract the data from this list and load it into an Excel workbook.

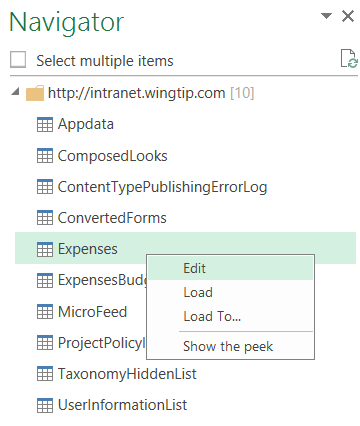
1. Return to Microsoft Excel ensure the workbook named **PowerQueryLab.xslx** you created in the previous exercise is still open.
2. Create a new query from the **Expenses** list in the SharePoint site at <http://intranet.wingtip.com>.
   1. In the Excel application window, navigate to the Power Query tab in the ribbon.
   2. Locate the **From Other Sources** button inside the **Get External Data** group.
   3. Drop down the **From Other Sources** button and click the **From SharePoint List** command.



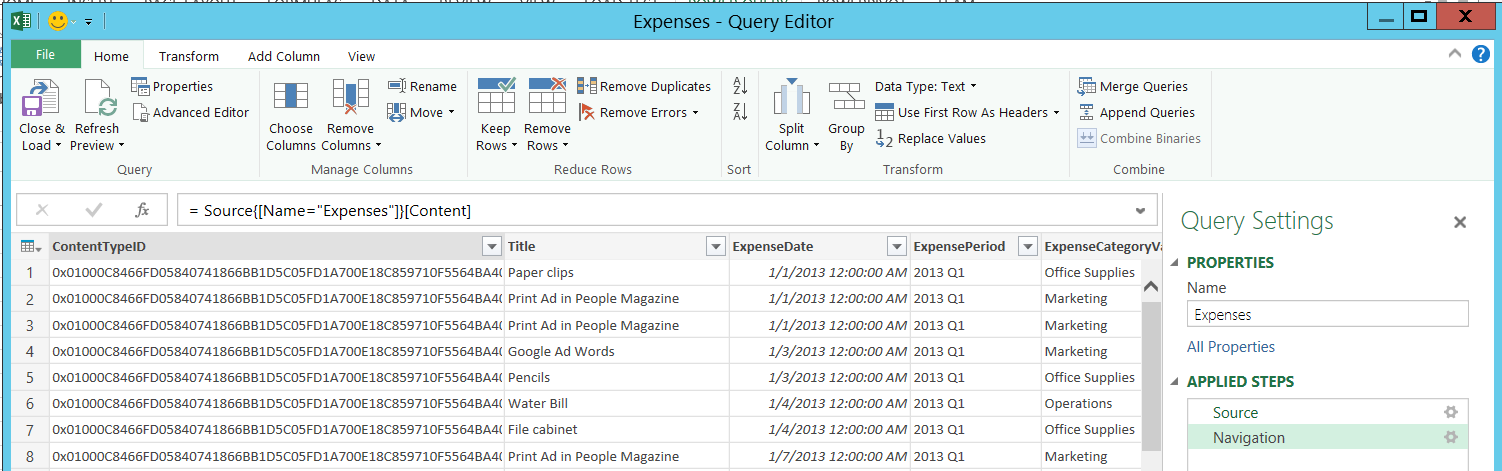
* 1. When promoted with the **Microsoft SharePoint Lists** dialog, enter the URL of <http://intranet.wingtip.com> and click **OK**.



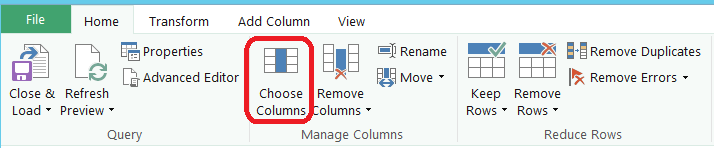
* 1. At this point, Power Query should display the **Navigator** pane which shows a list of SharePoint lists which are all accessible in the current site as shown in the following screenshot. Right-click on the **Expenses** list and select the **Edit** command to create a new query based on the SharePoint **Expenses** list which is opened in a new Query Editor window.



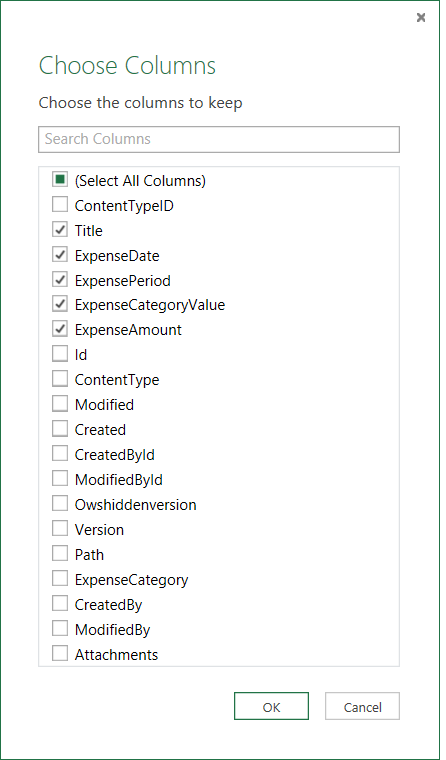
* 1. You will notice that a SharePoint list contains several columns that you usually want to hide such as the **ContentTypeID** column. Over the next few steps you will remove unwanted columns.



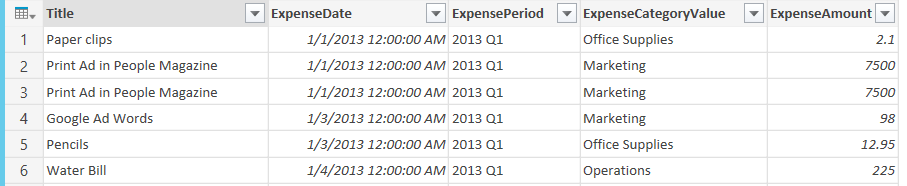
* 1. Click the **Choose Columns** button in the **Manage Columns** group of the **Home** tab to display the **Choose Columns** dialog.



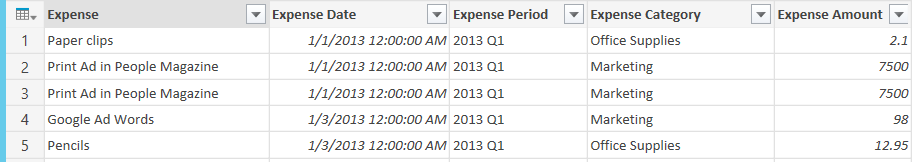
* 1. On the **Choose Columns** dialog, begin by uncheck the (Select All Columns) checkbox to unselect all columns. Next, select the columns you want to appear in the query results which are **Title**, **ExpenseDate**, **ExpensePeriod**, **ExpenseCategoryValue** and **ExpenseAmount**. When your selections match the following screenshot, click **OK** to save your changes.



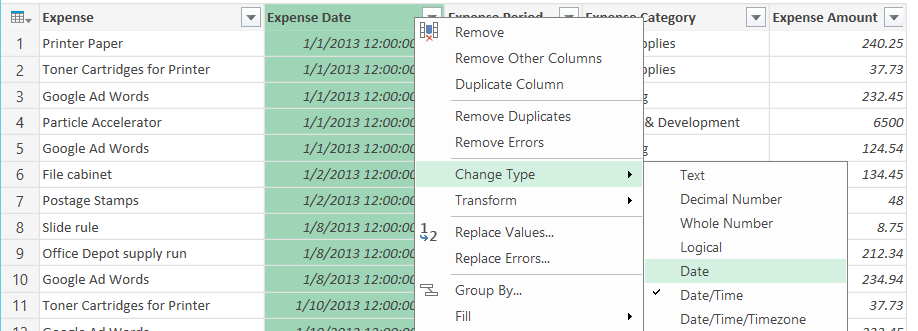
* 1. Note that the column names in the current query results can be improved to make them more readable.



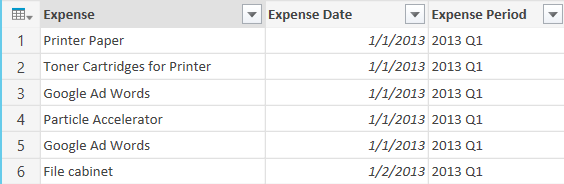
* 1. Change the name of the following columns.
     1. Update the name of the **Title** column to **Expense**.
     2. Update the name of the **ExpenseDate** column to **Expense Date**.
     3. Update the name of the **ExpensePeriod** column to **Expense Period**.
     4. Update the name of the **ExpenseCategoryValue** column to **Expense Category**.
  2. When you are done changing column names, the query results should appear as the screenshot below.



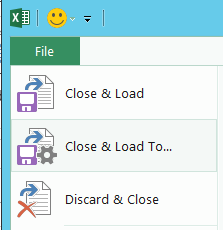
* 1. Right-click the header of the **Expense Date** column and select the **Change Type > Date** command.



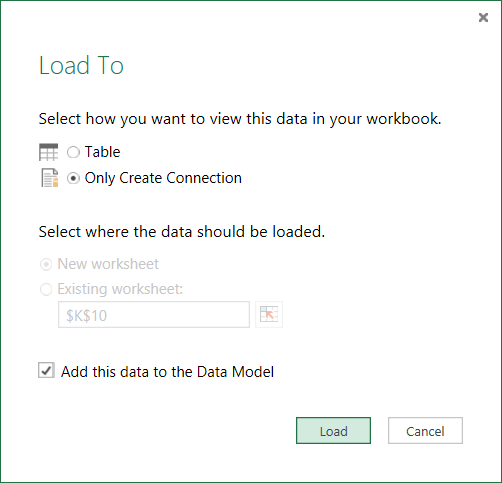
* 1. After change the **Expense Date** column to the type of **Date**, you should now see its column values are displayed as pure dates with no time element.



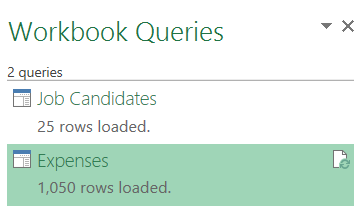
* 1. Now it's time to save the query and configure its output behavior. Begin by dropping down the **File** menu in the upper left-hand corner of the Query Editor window and selecting the **Close & Load To…** command to display the **Load To** dialog.



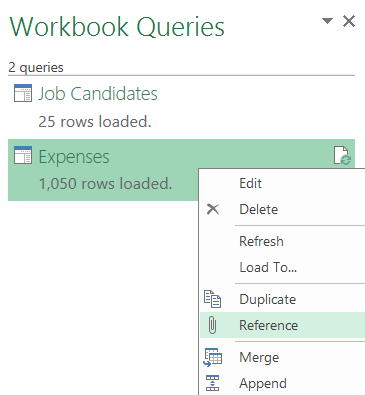
* 1. In the **Load To** dialog, select the option button for **Only Create Connection**. This prevents Power Query from loading the query output into a new table in a worksheet in the current workbook. Also make sure to click and add a checkmark to checkbox with the caption **Add this data to the Data Model**. This option tells Power Query to use the query output to create a new table in the Excel data model for the current workbook. When **Load To** dialog matches the one shown in the following screenshot, click the Load button to load the query output into the data model.



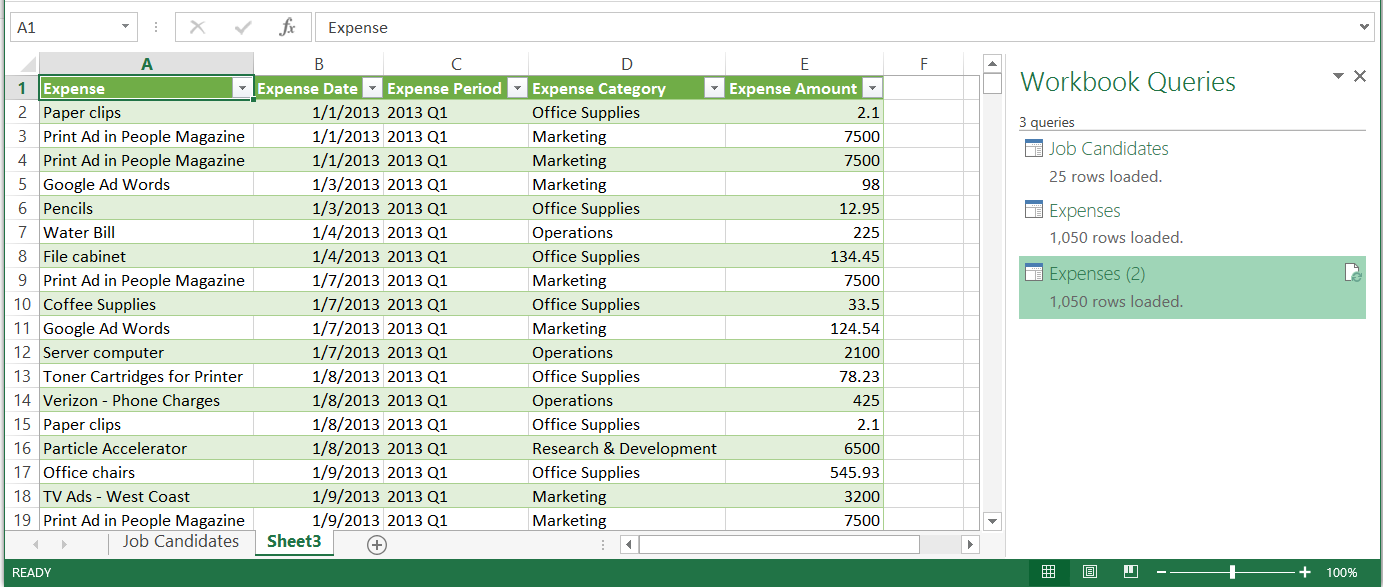
* 1. Examine the Workbook Queries view and verify you can see the **Expenses** query in addition to the **Job Candidates** query.



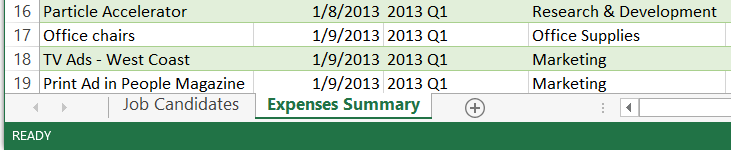
1. Create a new query for **Expenses Summary**.
   1. Right-click on the **Expenses** query in the **Workbook Queries** view and select the **Reference** command which will create a new query which use the **Expenses** query as its source.



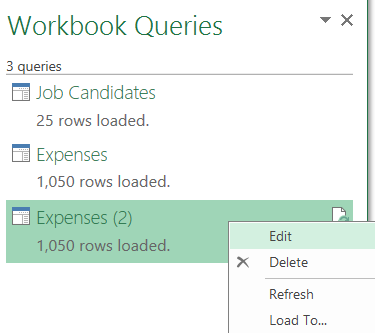
* 1. When you select the **Reference** command on a query named **Expenses**, Power Query will create a new query with a name such as **Expenses (2)**. It also adds a new worksheet where it can load the query results. The new worksheet will have a generic name such as **Sheet3**.



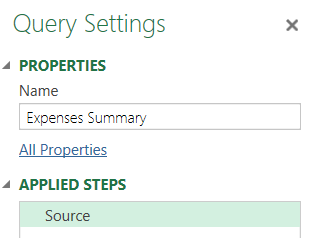
* 1. Update the name of the new worksheet that has just been created to **Expenses Summary**.



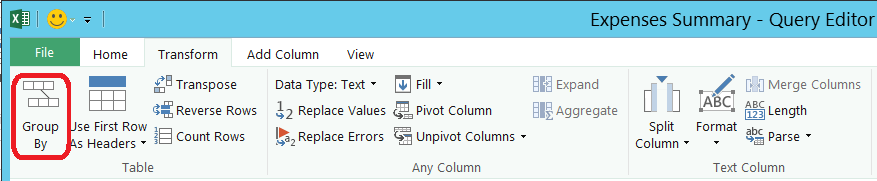
* 1. In the **Workbook Queries** view, right-click the new query and select the **Edit** command to open this query in the Query Editor window.



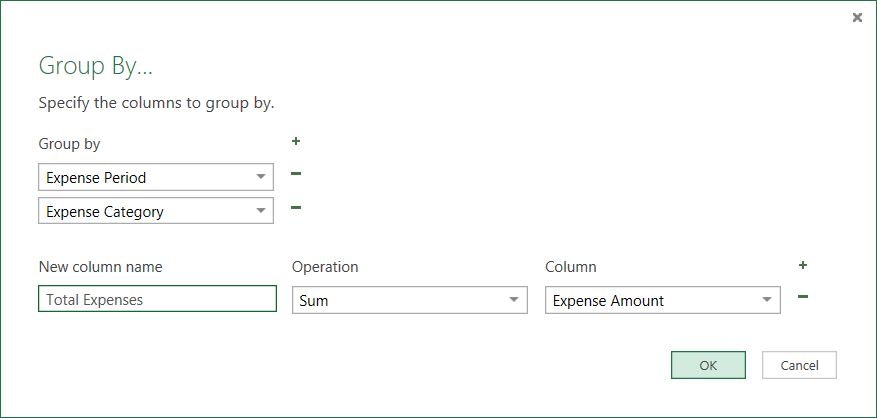
* 1. In the **Query Settings** view of the Query Editor window, change the Name property of this query to **Expense Summary**.



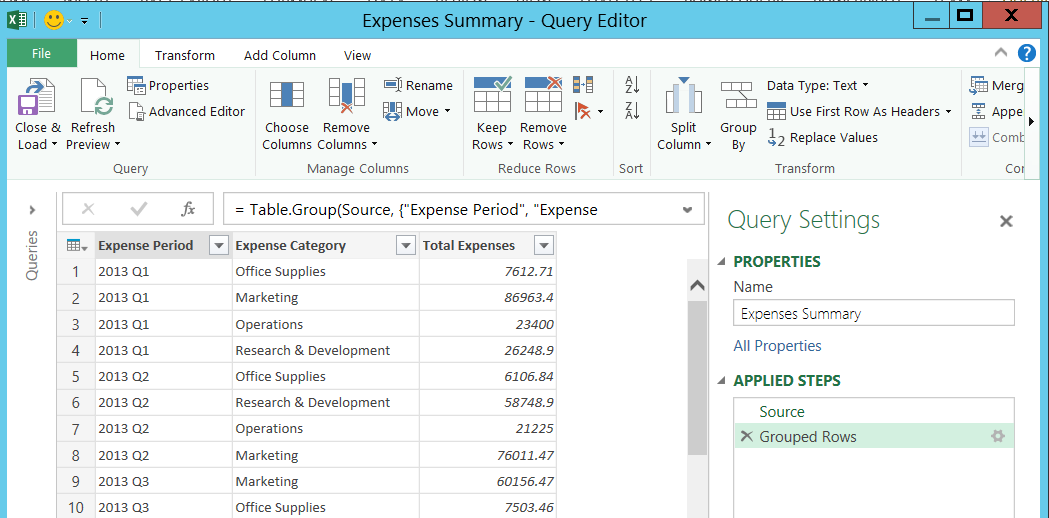
* 1. In the Query Editor window for **Expense Summary**, navigate to the **Transform** tab in the ribbon and click the **Group By** button to display the **Group By** dialog.



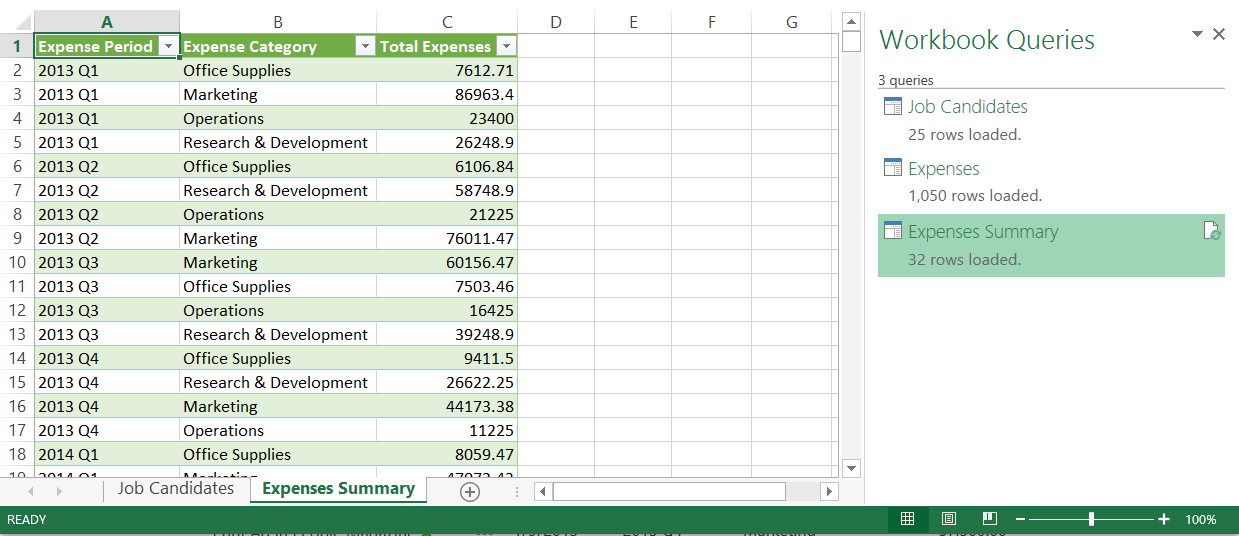
* 1. Fill out the **Group By** dialog to match the one shown in the following screenshot. More specifically, you should first group by **Expense Period** and then group by **Expense Category**. Add a new aggregated column named **Total Expenses** which will display the aggregated sum of **Expense Amount** column which is being rolled up. Click OK when you are done.



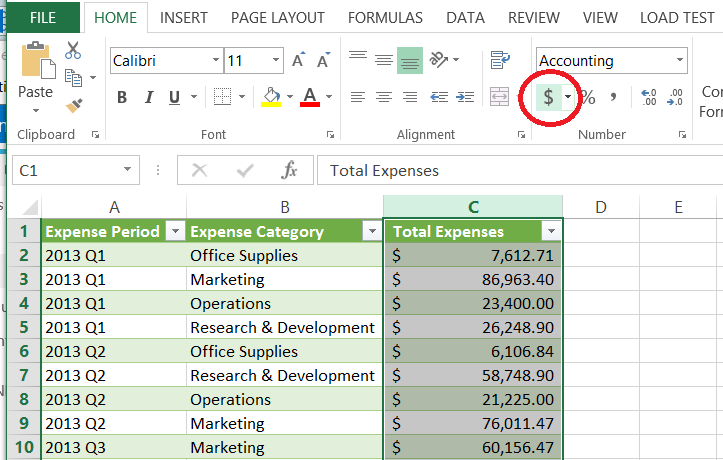
* 1. At this point, the results view in the Query Editor window should show three columns which are **Expense Period**, **Expense Category** and **Total Expenses**.



* 1. Drop down the **File** menu at the top left of the Query Editor window and select the **Close & Load** command. This will close the Query Editor window and reload the query output into the Expenses Summary worksheet.



* 1. The final step is to format the worksheet so that Total Expenses column shows as a currency value. Click on the Excel workbook column header for the C column to select the entire column. next click on the $ in the ribbon to apply currency formatting to all the **Total Expense** value.



1. You have now reached the end of this exercise.