



Paula Guilfoyle CPA has been an online Excel and Power BI instructor with TheExcelClub.com since 2010. During this time over 18,000 learners across 166 countries have taken her courses.

# Excel's BI Tools: An Introduction to Excel's Power Query (Get and Transform Data)

Paula Guilfoyle's second article on Excel brings you through how to transform data into a usable format.

In the last edition of Accountancy Plus you were introduced to the use of Power Pivot. Power Pivot was the first of the Business Intelligence Power Tools introduced by Microsoft. Power Pivot, on its own proves to be an extremely powerful and useful tool, allowing you add data to a model and create relationships across a table of data for reporting. But when you are looking at Business Intelligence data can come from many sources and in many forms. If you really want to analyse data in Excel then you need to get the data and transform the data into a useable format.

So what is a useable format in Excel terms? The descriptive word would be tabular data. It doesn't matter if this data is text, or numerical data. To analyse data in a pivot table or power pivot report then you need to have your data in a tabular format.

Working in Excel it's easy enough to make sure your data is in the right format, but sometimes even Excel data is not in the correct format. Imagine how the data could be structured when it is an external data source, such as data from the internet.

Microsoft have made all of this rather simple for us. In 2013 the Excel add in Power Query was released. In Excel 2016 this becomes a standard feature of Excel known as Get and Transform Data.

Power Query has masses of Power, with an interface that allows non-programmers transform data like never before. And if you are a programmer, then you will be delighted to hear that behind the user interface Power Query has its own language know as M. However as Excel is a self-service system, Microsoft have been kind enough to ensure most of the transformations can be carried out by the Excel user and not a programmer.

It's hard to show the capabilities of Power Query in just one short article, but for now we will look at how Power Query can be used to combine a folder of CSV files. Saving you massive amounts of time on re-occurring tasks.

### Transforming Data with Power Query

Remember our ice-cream distributor from the last article. Well with some system improvements, they are no longer keeping track of sales in Excel, but now they are using a system. Each month the accountant downloads a CSV file containing the sales data and saves the file in a folder set up just for these files. But we all know that CSV files need a little work, even in plain old excel. So let us look at setting up a Power Query that will connect to a folder of CSV files and transform the data into a useable format.

In Power Query (or Get and Transform Data in Excel 2016) select Get external Data from Folder, see Figure 1.

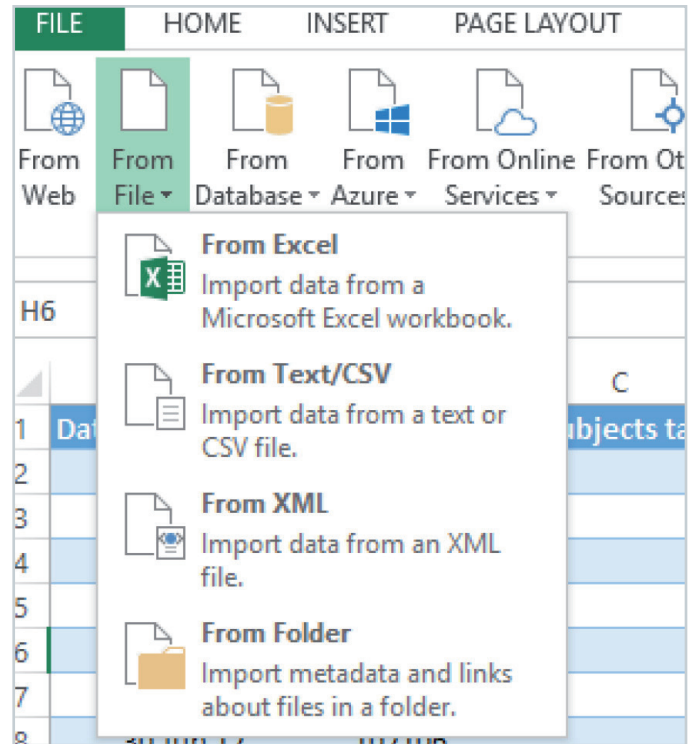


Figure 1

Then navigate to the folder in which the CSV files have been saved. Once the folder has been selected, a new screen showing details of each file in the selected folder will appear. From here, select Combine, and select Combine and Edit, see Figure 2.

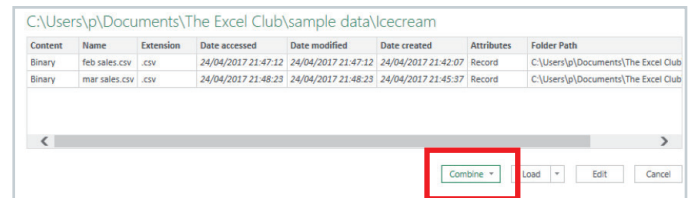


Figure 2

Power Query runs through the data and returns a suggested separation for the data in the files within the folder. Most often this suggestion is correct, however if it is not, changes to the delimiter can be made manually to get the correct separation. Once this is correct, ok is then selected, see Figure 3.

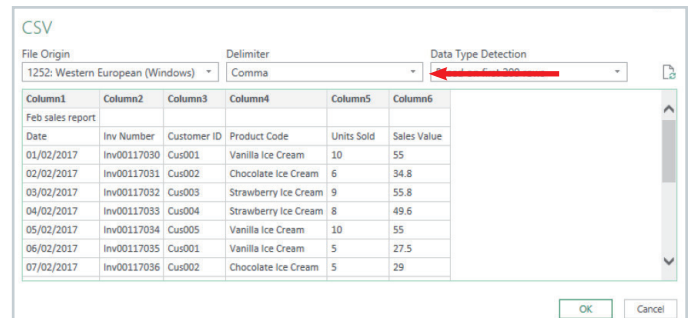


Figure 3

► Continued on Page 31

▶ Continued from Page 32

Power Query editor will now open. The place where all the magic happens and all the transformations take place. On the right of the editor, Query settings, is where you will find all of the transformation steps that you have applied so far. On the top we have some ribbons just like excel, and down the left you will find the Data Queries, see Figure 4.

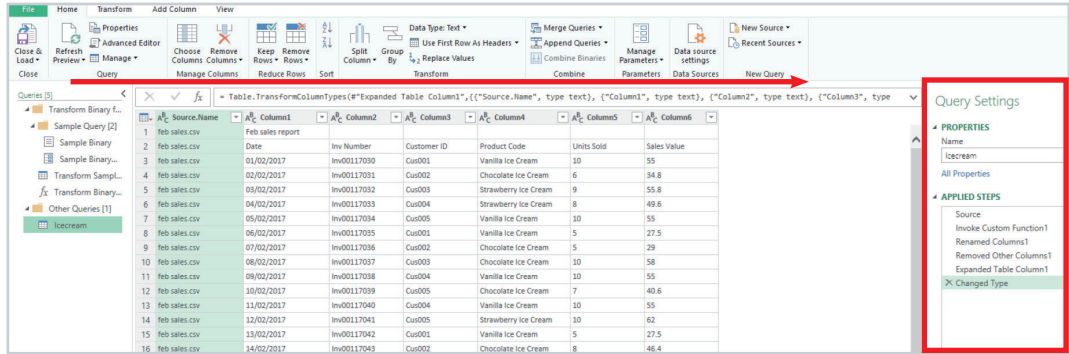


Figure 4

The steps we have taken so far are the automatic steps to combine binary's (or files) within a folder. Now we need to do a little work on the data.

We have imported two files of the same format and we can see that Column2 has no data in row 1. This was a header row in the CSV file. Let's remove this by first selecting Column 2, and in the filter remove the check on Blank Rows, see Figure 5.

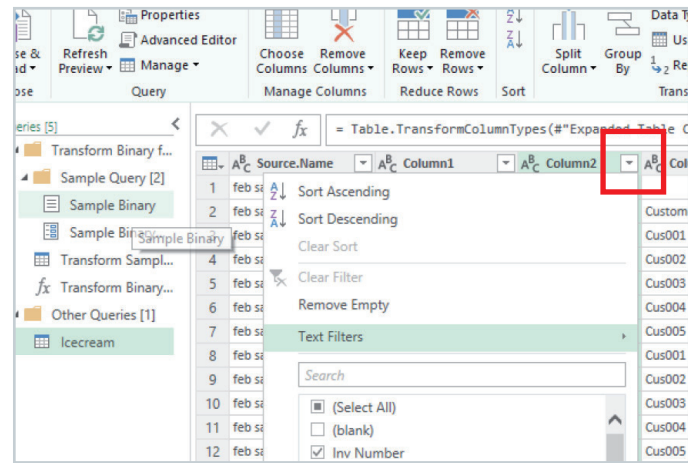


Figure 5

The next step would be to add some column headers, as at the moment we have Source.Name and then Columns 1-6. In the Home Ribbon you can now select Use First Row as header, see Figure 6.

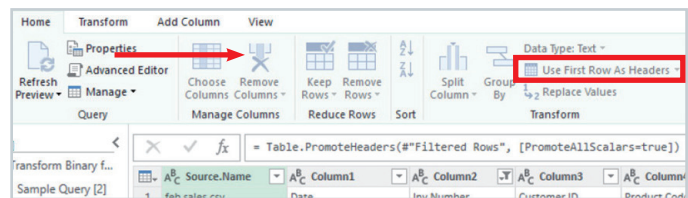


Figure 6

The column that was originally called Source.Name is now Feb sales.csv. We don't really need this column so we can remove it by selecting the column and in the Home Ribbon, select Remove Columns.

Scrolling down our data now to see if there are any further transformations, we see that there is a row that contains the same data as the column headers. This is because in each CVS file there is a header row. We can remove this by selecting any of the filters on the column headers and unchecking the row we do not need, see Figure 7.

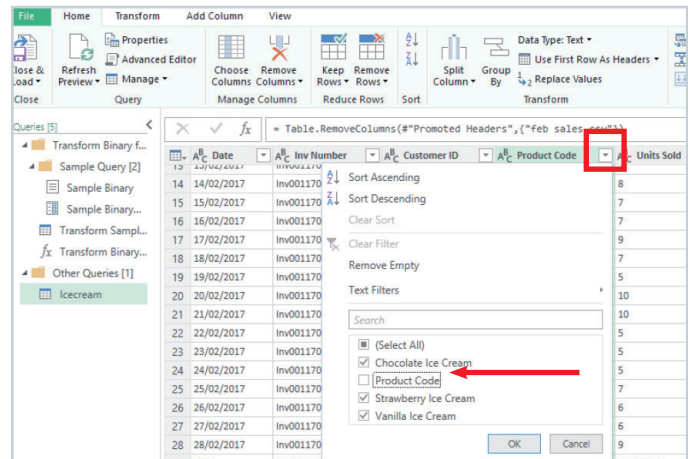


Figure 7

Now that we have our data transformed, we can select either Close or Load, which will load the transformed data to a table in Excel or Close and Load to, which will load the data into the data model ready to be used in Power Pivot for further analysis, see Figure 7.

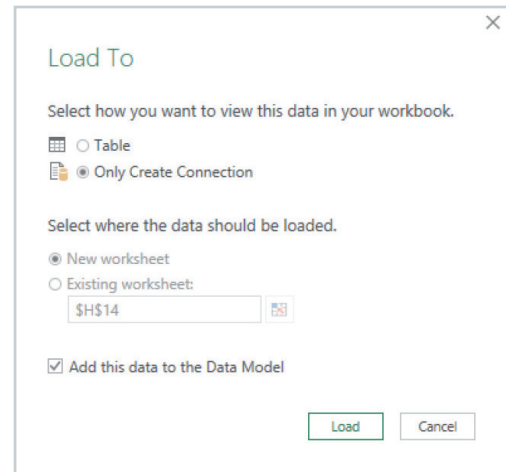


Figure 7

## Back to DAX

In the last article we looked at some simple DAX calculations. DAX rocks. Now we have multiple months of data in our data model we could analyse our data by time periods and use some of the DAX Time Intelligence Functions. Last time we only had one period of data which we sliced by customer. Using the same data model we can now slice by both customer and time period.

But for DAX Time Intelligence Functions to work, you need to have a Date table in your model. For a Date table to be complete it must list each and every one of the days of the years in the model.

Let's say you have a shop that does not open on Saturdays and Sundays. It would not have sales for these days, but the Date table would still need to include these dates. There should be no gaps at all in the Date table.

A Date table can be created in Excel and added to the model as we have seen in the first article. A Date table can also be created in Power Query with a little knowledge of M (or a copy of the script!!) And a Date table can be created using DAX.

Excel 2016 includes a new DAX function to help create a Date table. Select Power Pivot and select manage to open up Power Pivot. Then select New table from the Get Data tab. In the formula bar enter

```
=CALENDARAUTO([monthofyearend])
```

If the year end differs from the calendar year end, you can enter the month number for your year end. This will now create a new table with all of the required dates in the model.

Let's just add two further calculated columns to this table. The first will have the DAX expressions

```
= MONTH('Date'[date])
```

This will return the month number for each month. Next we will create a column using the expression

```
= FORMAT(MONTH('Date'[date]),"MMMM")
```

to return the month name. Now we have created a column which we can use as a sort column and we have also added the month by name which we can use to slice and pivot our data.

The last step in creating the Date table is to tell DAX that it is a DATE table. To do this, in Power Pivot, under the Design tab, select 'Mark as Date Table' see Figure 8.

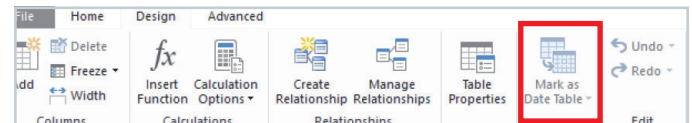


Figure 8

In the relationships tab we can now create a relationship between the Sales table and the newly created Date table.

Let's now pivot our data by creating a new pivot table, adding the month to the Rows of the Pivot table and the Customer to the Columns. Now we have created a report containing all of the month's sales, split by customer.

## The Magic of Power Query

Now here is where things get really exciting. Yes we have just created a report and sliced it very quickly, something that would have taken a lot of work in Excel, but check this out! Once you save this Excel file, with the query, any new CSV you add to the folder will also be transformed and added to the model with a simple refresh of the data. There is no need to carry out the same steps month on month.

Prior to Power Query, this was only possible with VBA coding. Now you can save hours without the need to know VBA.