

Tasks

Task 1 - Exploring the M Query library with #shared

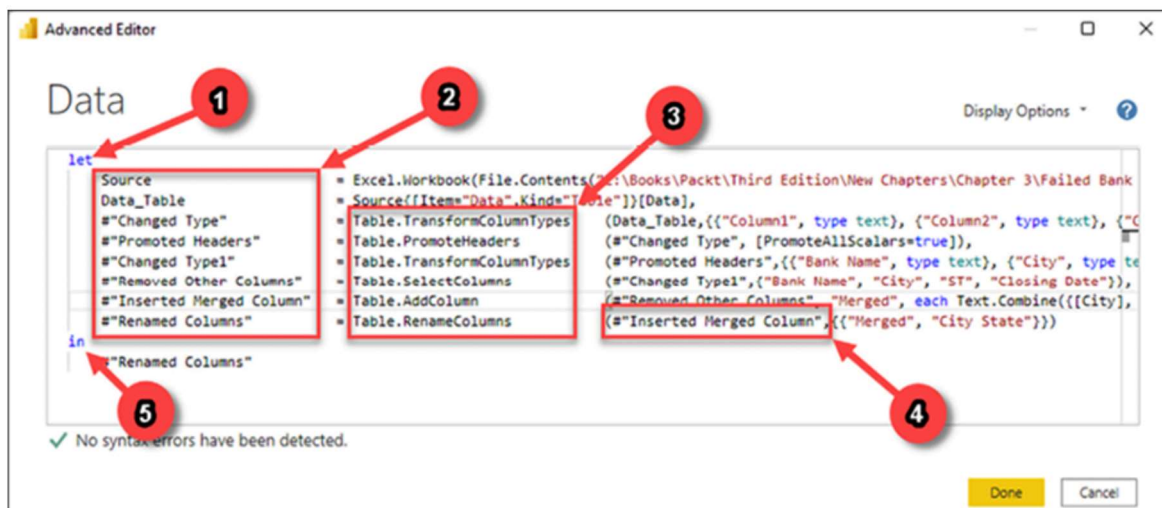
The *Power Query Editor* is the user interface that is used to design and build data imports. However, you should also know that every transform you apply within this editor is actually, quietly and behind the scenes, writing an **M** query for you. The letter *M* here is a reference to the language's data mash-up capabilities.

For simple solutions, it is unlikely that you will ever need to even look at the *M* query that is being written, but there are some more complex cases where it's helpful to understand how to read and write your own *M*. For the purposes of this course, covering just the *Power BI* essentials, you will learn how to find the *M* query editor within your solution and then understand how to read what it is doing for you.

For the purposes of this example, you can open up any previously built example; however, the screenshot used here is from the very first example in this chapter on basic transforms:

Using any Power BI solution you have designed, launch the Power Query Editor.

On the *Home* ribbon, select *Advanced Editor* to see the *M* query that has been written by the user interface. Figure below shows an example of what your *Advanced Editor* might show:



This query has been formatted to make it easier to read. Let's review the key elements that are present here:

1. **The let expression:** Encapsulates a set of values or named expressions to be computed.
2. **Named expressions or variables:** The name given to a set of operations in a step. These names can be anything, but you should note that if you wish to have a space in the name of a step then it must be surrounded by #"". For example, if I wanted something to be called *Step 1*, then I would have to name an expression *#"Step 1"*. If a space is not required in the name of your step, then double quotes are not required.
3. **M functions:** The operations that are used to manipulate the data source.
4. **Prior step reference:** The *M* query language generally executes its functions as serial operations, meaning each operation is executed one after the other sequentially. You can see this when you look at a query because each call to an *M* function always references the prior-named expression, to pick up where it left off.

5. **The in expression:** Oddly, the in expression is actually a reference to what the query will output. Whichever named expression is referenced in the in expression will be what is returned in the *Power Query Editor* preview.

It is important to realize that *M* is case-sensitive. That means if you ever make a change to a query or write one from scratch, you should be careful because there is a difference between “a” and “A.”

Exploring the M Query library with #shared

As mentioned previously, this course will not dive deep into writing your own *M* queries since that would be far beyond the essentials of *Power BI*. However, there is a great method for exploring the *M* functions that are available, and how to use them. Within the *Power Query Editor*, you can use the *#shared* function to return documentation on every available function in the *M* library. Let's walk through how you can leverage this tool:

Step 1: In a new instance of *Power BI Desktop*, select **Get data** and then choose **Blank Query**. This will launch the *Power Query Editor* with an empty formula bar waiting for you to provide your own *M*.

Step 2: In this formula bar, type "`= #shared`", then hit **Enter**. Remember that **M** is case-sensitive so you must use a lowercase s when typing *shared*.

Step 3: This will return a list of all the available M functions. By selecting the cell that has the hyperlink text of a certain function, you can see documentation on how to use each function. Figure below shows an example of this:

✕	✓	fx	= #shared	▼
Table.MatchesAnyRows	Function	▲		
Table.Partition	Function			
Table.Range	Function			
Table.RemoveRows	Function			
Table.Repeat	Function			
Table.ReplaceRows	Function			
Table.ReverseRows	Function			
Table.HasColumns	Function			
Table.PrefixColumns	Function			
Table.ColumnsOfType	Function			
Table.AddColumn	Function			
Table.DuplicateColumn	Function			
Table.FillUp	Function			▼
Table.RemoveColumn	Function			

End-of-Exercise

Olet suorittanut 100 % oppitunnista

100%

◀ Exercise 6 - AI Insights

Siirry...

Lesson 3 Quiz ►

Olet kirjautunut nimellä Janne Bragge. (Kirjaudu ulos)
PowerBI

Suomi (fi)

Deutsch (de)

English (en)

Français (fr)

Suomi (fi)

Svenska (sv)

Hanki mobiilisovellus

