Advanced Power BI (Data)

Sample manual - first two chapters

Wise Owl Training

TABLE OF CONTENTS (1 of 5)

TABLE OF CONTENTS (2 of 5)

TABLE OF CONTENTS (3 of 5)

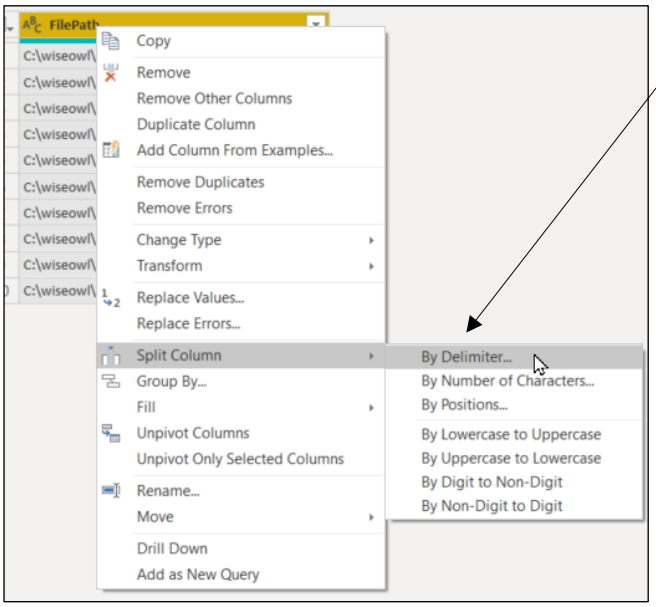
TABLE OF CONTENTS (4 of 5)

TABLE OF CONTENTS (5 of 5)

CHAPTER 1 - MANIPULATING COLUMNS

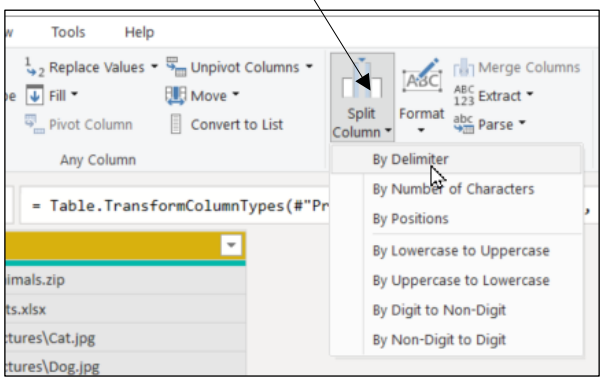
1.1 Splitting Columns

You can split columns using either the right mouse button menu or the ribbon:



Either right-click on a column that you want to split and choose one of these options …

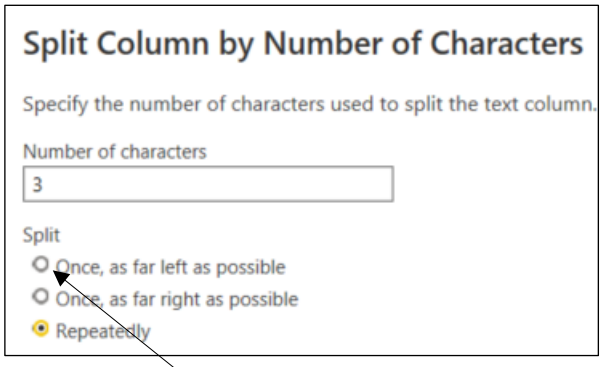
… or choose one of the options from the **Home** tab of the Query Editor ribbon.



Splitting by Number of Characters

If you know how many characters you want to extract, you can use this method:

Product Code	Sales
MPB-4242	24.49703694
HQC-0233	44.9529665
ZND-9682	91.64723297
PGL-4142	30.60766405
ESQ-9921	29.87047969



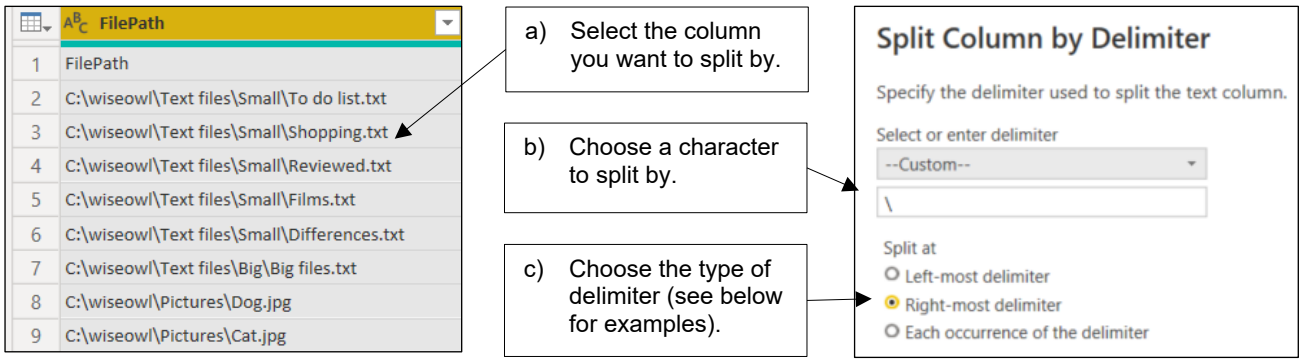
a) Select the column that you want to split, and choose one of the options shown above to split it,

b) Choose to take the first N characters, the last N characters or a series of N-character chunks. To get the first 3 letters in this case we would choose the first option (**Once, as far left as possible**).

For the above example, you could instead split the product code by a delimiter as shown overleaf (in this case, the - character) to get the letters and numbers separately.

Splitting by Delimiters

For the example below, you might want to extract the file name or file path, so you'd probably split at the last `\` character:



Here are the first couple of rows you'd get for each of the 3 **Split at** options above:

Option	Results	Notes
Left-most delimiter	FilePath.1: C:, C: FilePath.2: wiseowl\Text files\Small\To do list.txt, wiseowl\Text files\Small\Shopping.txt	Split by the first backslash
Right-most delimiter	FilePath.1: C:\wiseowl\Text files\Small (×3) FilePath.2: To do list.txt, Shopping.txt, Reviewed.txt	Split by the last backslash
Each occurrence of the delimiter	FilePath.1: C: FilePath.2: wiseowl FilePath.3: Text files FilePath.4: Small FilePath.5: To do list.txt, Shopping.txt, Reviewed.txt	Split by every backslash

Splitting by Positions

Use this method to extract a substring of text:

	Product code	Price
1	VPW-85/L9-50	84.8176205
2	COS-84/G6-98	94.1651118
3	MZZ-45/A3-22	17.25046236
4	VWO-51/P3-17	79.03115968
5	NHS-21/I3-49	88.90362863

To extract the two-digit number after the first dash from each product code you could use this transform:

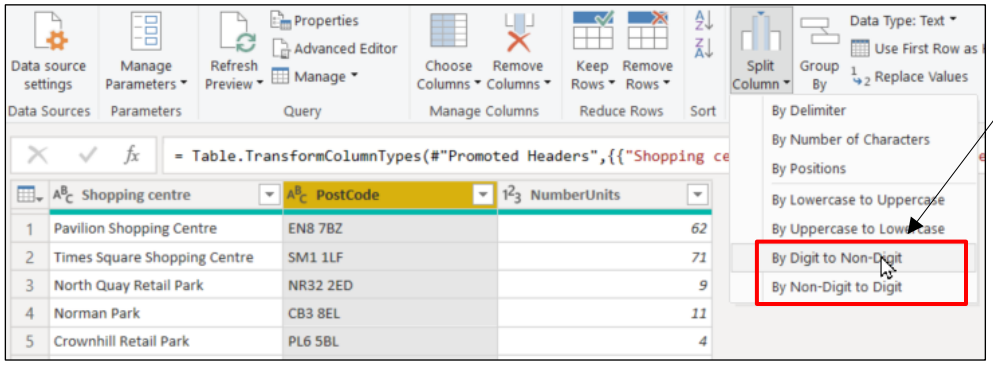
This would give these 3 columns (everything from character 0 to 3, 4 to 5 and 6 onwards).

	Product code.1	Product code.2	Product code.3
1	VPW-	85	/L9-50
2	COS-	84	/G6-98
3	MZZ-	45	/A3-22
4	VWO-	51	/P3-17
5	NHS-	21	/I3-49

*As with most things in Power BI, characters are numbered from 0, not 1. To omit the first column you could just put **4, 6**, to start extracting characters at position 4.*

Splitting by Change in Case or Character Type

Codes are often divided into a predictable combination of letters and numbers (or of upper and lower case characters). Here's how to split up some well-behaved UK postcodes:



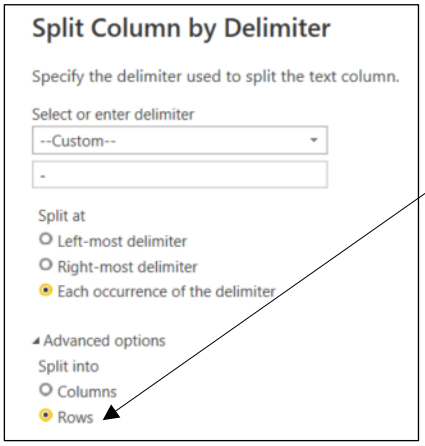
By repeated application of these two transforms, for example …

	PostCode.1.1	PostCode.1.2	PostCode.2	PostCode.3
	EN	8	7	BZ
	SM	1	1	LF
	NR	32	2	ED
	CB	3	8	EL
	PL	6	5	BL

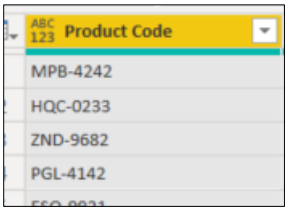
… you could split each postcode into the letter(s) and number(s) before the space and the letter(s) and number(s) after it.

Splitting into Rows

Sometimes you may want to create one row for each constituent part of a string of text (although it's hard to think of an example!):

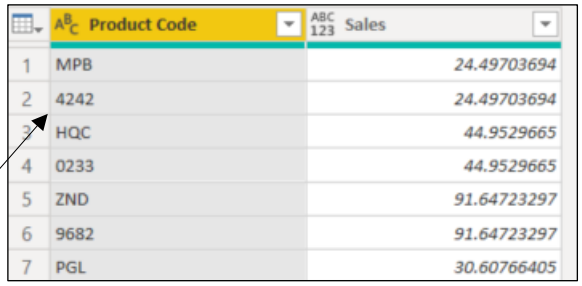


If you expand the advanced options when splitting a column and choose to generate extra rows, not columns for this example …



	Product Code
	MPB-4242
	HQC-0233
	ZND-9682
	PGL-4142

… you'll get multiple rows for each original one (notice that the values of the other fields will be duplicated for each of these rows, making the data hard to interpret in most cases).

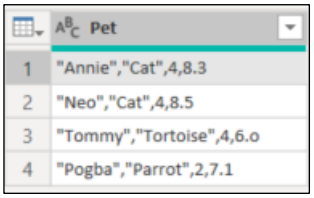


	Product Code	Sales
1	MPB	24.49703694
2	4242	24.49703694
3	HQC	44.9529665
4	0233	44.9529665
5	ZND	91.64723297
6	9682	91.64723297
7	PGL	30.60766405

Retaining Quotation Marks

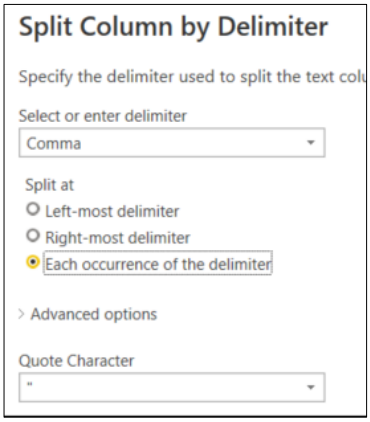
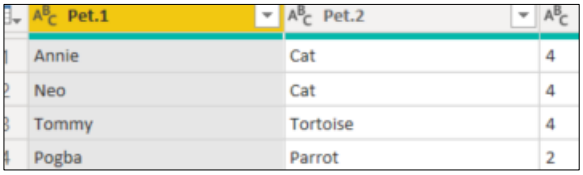
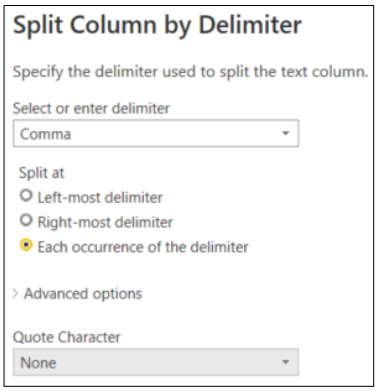
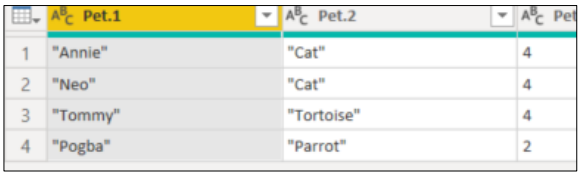
When splitting text, Power Query will automatically remove any quotation marks, but you can turn this option off as shown below.

The examples below use this as an example source query.



Pet
"Annie","Cat",4,8.3
"Neo","Cat",4,8.5
"Tommy","Tortoise",4,6.o
"Pogba","Parrot",2,7.1

Here are the two options:

Option	Dialog box	Results for our example
Remove quote marks		
Keep quote marks		

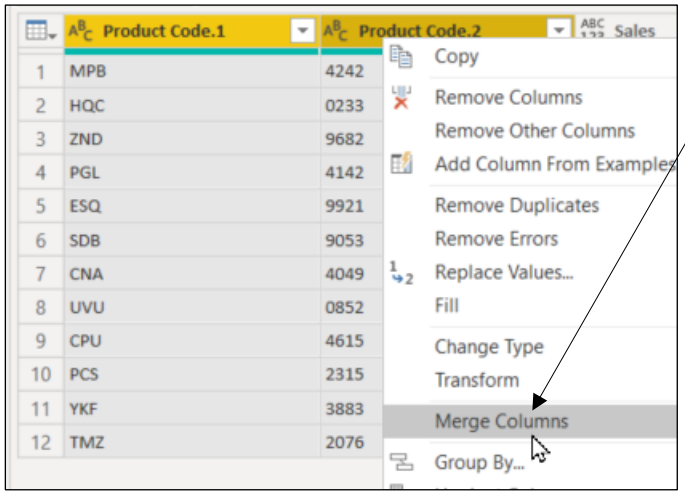
Pet.1	Pet.2	
Annie	Cat	4
Neo	Cat	4
Tommy	Tortoise	4
Pogba	Parrot	2

	Pet.1	Pet.2	
1	"Annie"	"Cat"	4
2	"Neo"	"Cat"	4
3	"Tommy"	"Tortoise"	4
4	"Pogba"	"Parrot"	2

There doesn't seem to be any option for automatically removing single quotation marks (or any other characters).

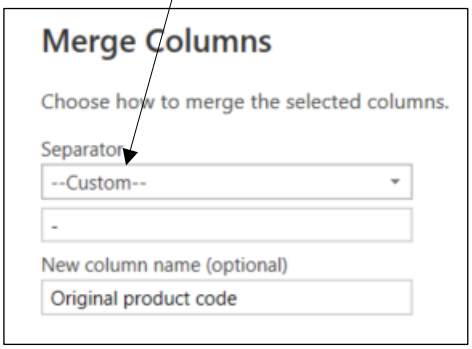
1.2 Merging Columns

The opposite of splitting columns is *merging* them. Here's an example:



a) Suppose you want to join the parts of the product code that we split earlier in this chapter back together again! To do this, first select the columns that you want to join together, right-click on them and choose **Merge Columns**.

b) Choose the glue you want to use to join the values together for each row (here we've gone for a dash).



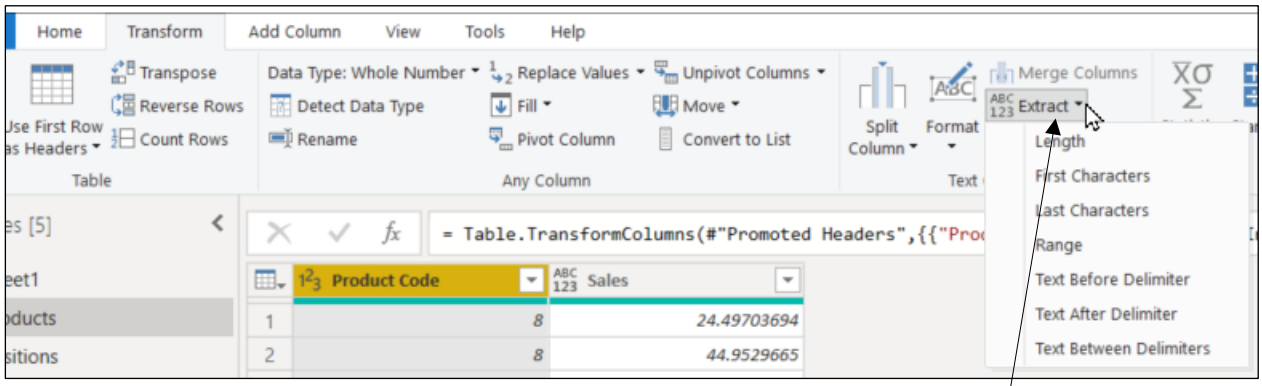
c) Power Query has combined the columns together into a single column.

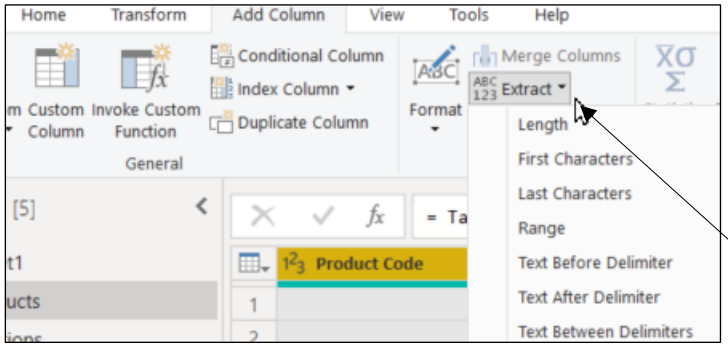
You're not limited to just two columns: you can merge as many columns as you like into a single one.

1.3 Extracting Data

Replacing or Adding Columns

The **Extract** tool lives in two places:



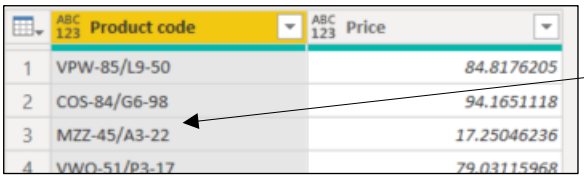


If you choose this version on the **Transform** tab of the ribbon you will replace a column.

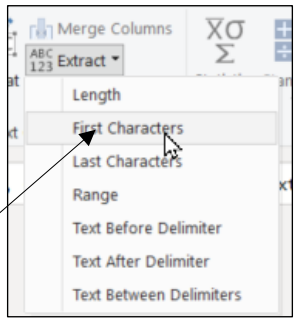
If you choose this version on the **Add Column** tab of the ribbon you will keep your existing column and add a new one to it.

An Example

The example below would extract the first 3 characters from a product code:



a) Select the column (or columns) from which you want to extract data.



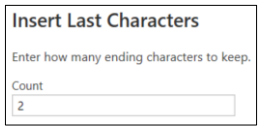
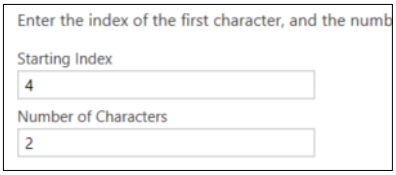
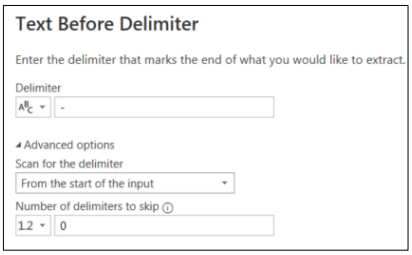
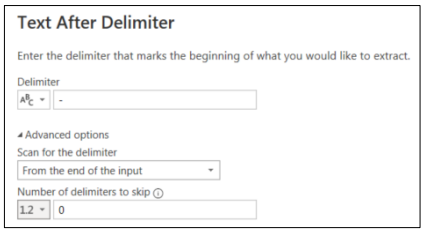
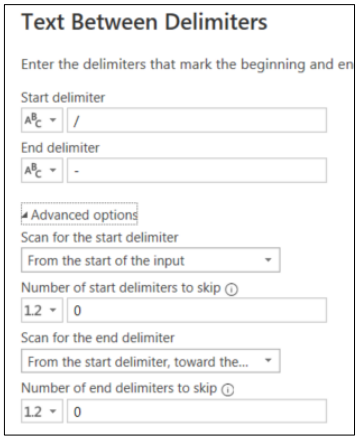
b) Here we're choosing to get the first N characters, replacing the current column.

c) Specify how many characters you want to extract (here we've gone for the first 3).

d) Power Query will replace your column with the first 3 letters for each.

The Possible Options

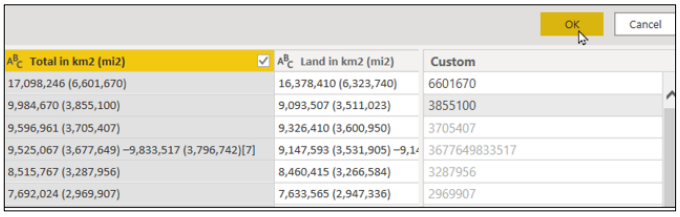
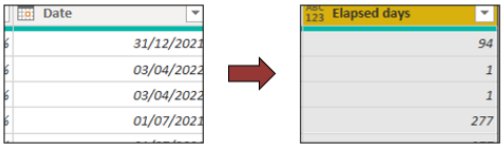
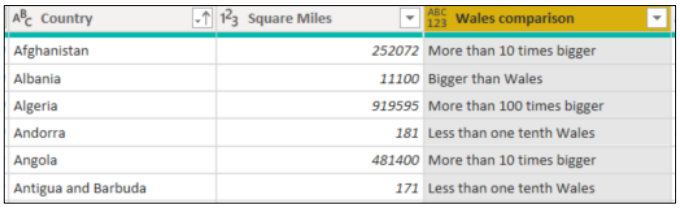
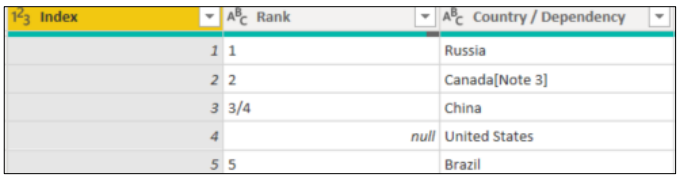
Here's what the options would give for a field containing the product code **VPW-85/L9-50**:

Option	Choices	Results	Notes
Length	None	12	This is a quick way to find how many characters there are for each value in a column.
First Characters		VPW	The (in this case) 3 left-most characters.
Last Characters		50	The (in this case) 2 right-most characters.
Range		85	The 2 characters starting at position number 4 (remember that characters are numbered from 0, not 1).
Text Before Delimiter		VPW	Notice that you can start scanning for a delimiter from the left or right with these options, and also choose not to take the first delimiter you find.
Text After Delimiter		50	
Text Between Delimiters		L9	This powerful option allows you to pick out text between any two (possibly different) characters. You can even do things like pick out the text between the second dash and fourth backslash from the end.

CHAPTER 2 - CREATING COLUMNS

2.1 Ways to Create New Columns

Rather than creating new columns using DAX in Power BI, there are a variety of ways of creating them in Power Query, as shown in this chapter:

Method	What it does	Example
Column from Examples	Use AI to guess what formula you want to apply to all rows of a table based on one or two sample training values.	
Built-up Columns	Uses built-in Power Query transforms in sequence to produce complicated effects (you can then delete any intermediate columns created).	
Custom Columns	Returns the value - for each row of a table - of an expression using the M Power Query Formula language.	
Conditional Columns	Uses rules to divide rows up into a finite number of discrete categories.	
Indexing Columns	Applies numbering to the rows in a table.	

Columns that you create in Power Query will take up more space in your model (since all of the row values have to be pre-calculated when you load your data), but will run more quickly thereafter (since Power BI doesn't need to calculate DAX columns on the fly). In practice you're unlikely to notice the difference!

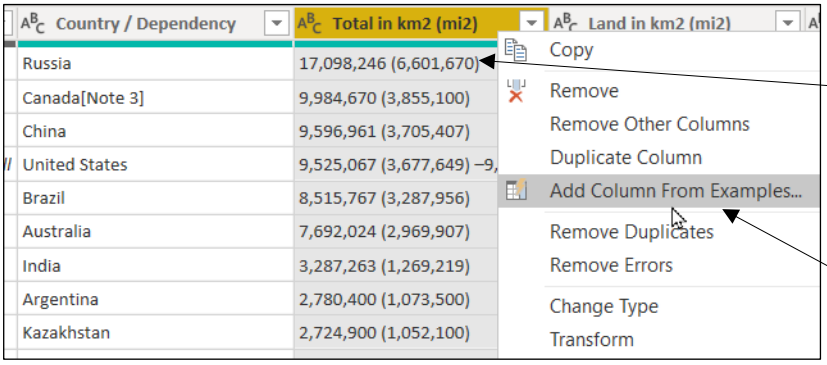
2.2 Columns from Examples

Power BI has got an excellent AI feature called *Columns from Examples*, which allows you to create formulae by typing in a few results. To create a column like this follow the numbered steps below.

There are very few examples in computing of an AI wizard that actually works, so kudos to Microsoft for investing so much money in creating a formula generation tool that gets things right so often.

Step 1 – Start the Feature

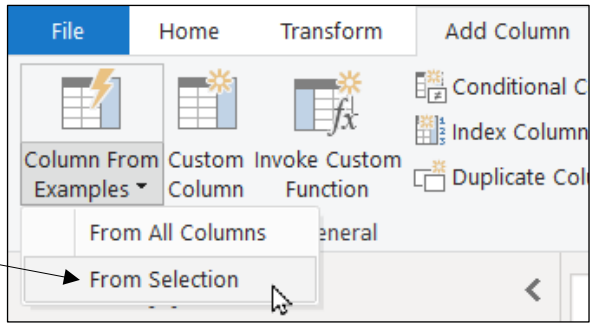
Begin by selecting the column or columns which contain the raw data from which Power Query can deduce the formula you want to apply, then right-click and choose this menu option:



For this example we want to show the total area of each country (the list comes from Wikipedia) in square miles. So for Russia this should give **6601670**, the figure in brackets.

Start by selecting the column or columns upon which Power Query should base its formula, then right-click and choose to add a column by showing some examples.

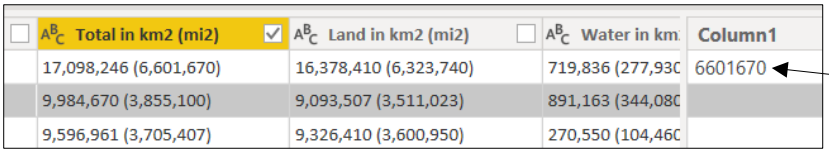
Note that you can also do this from the ribbon:



You can do the same thing by clicking on the drop-down next to the **Column From Examples** button on the **Add Column** tab of the Power Query ribbon.

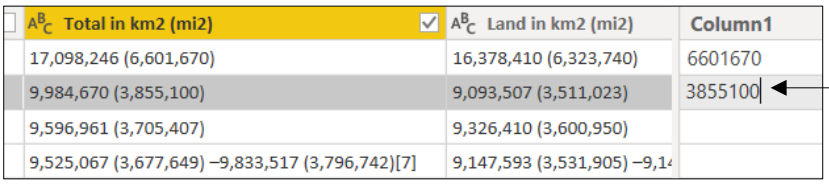
Step 2 – Show some Examples

You can now train the Power BI column formula generation algorithm:



Total in km2 (mi2)	Land in km2 (mi2)	Water in km	Column1
17,098,246 (6,601,670)	16,378,410 (6,323,740)	719,836 (277,930	6601670
9,984,670 (3,855,100)	9,093,507 (3,511,023)	891,163 (344,080	
9,596,961 (3,705,407)	9,326,410 (3,600,950)	270,550 (104,460	

a) Type in the value you would expect a Query Editor formula to return for your first row of data, then press Ctrl + ↵.



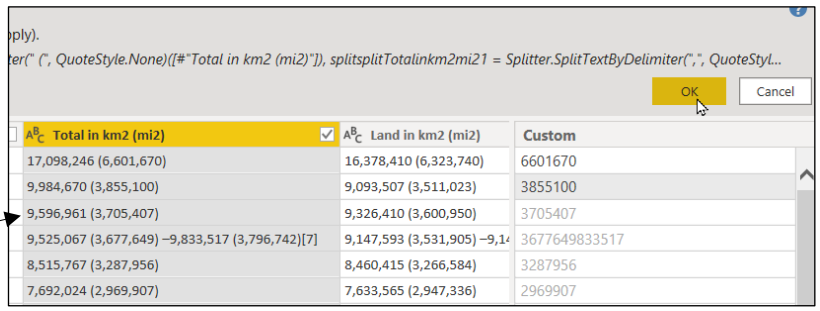
Total in km2 (mi2)	Land in km2 (mi2)	Column1
17,098,246 (6,601,670)	16,378,410 (6,323,740)	6601670
9,984,670 (3,855,100)	9,093,507 (3,511,023)	3855100
9,596,961 (3,705,407)	9,326,410 (3,600,950)	
9,525,067 (3,677,649) –9,833,517 (3,796,742)[7]	9,147,593 (3,531,905) –9,14	

b) If you don't get any results for the first value, try a second (and even a third) until Power BI gets what you're trying to do.

Step 3 – Confirm the Formula

If you're happy with what Power BI has created, select OK; otherwise, choose Cancel:

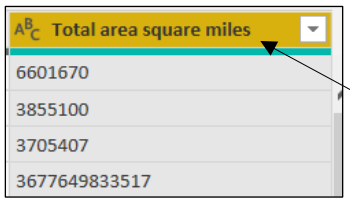
Incredibly, Power BI seems to have worked out exactly what you want to do (for the 4th row – the USA – the data is different, but we can work round this later).



Total in km2 (mi2)	Land in km2 (mi2)	Custom
17,098,246 (6,601,670)	16,378,410 (6,323,740)	6601670
9,984,670 (3,855,100)	9,093,507 (3,511,023)	3855100
9,596,961 (3,705,407)	9,326,410 (3,600,950)	3705407
9,525,067 (3,677,649) –9,833,517 (3,796,742)[7]	9,147,593 (3,531,905) –9,14	3677649833517
8,515,767 (3,287,956)	8,460,415 (3,266,584)	3287956
7,692,024 (2,969,907)	7,633,565 (2,947,336)	2969907

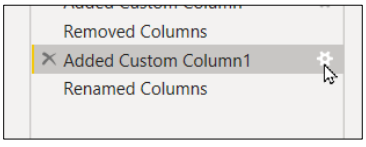
Step 4 – Review your Formula

You can now have a look at the M formula created for your new column:

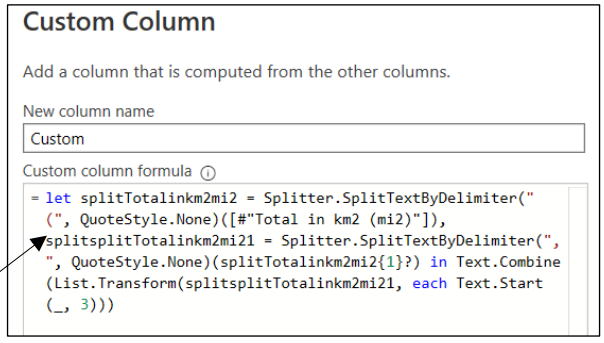


Total area square miles
6601670
3855100
3705407
3677649833517

a) Here we've renamed the column too. You can view the details of the transform step by clicking on the gear icon next to it:



b) This would show this formula, which is explained overleaf!)



Step 5 – Understanding your Formula

The big disadvantage of creating columns by example is that you may find it hard to understand (and amend) the resulting formula. For our example, here's our formula:

```
let splitTotalinkm2mi2 = Splitter.SplitTextByDelimiter(" (", QuoteStyle.None)([#"Total in km2 (mi2)"]),
splitsplitTotalinkm2mi21 = Splitter.SplitTextByDelimiter(",", QuoteStyle.None)(splitTotalinkm2mi2{1}?) in
Text.Combine(List.Transform(splitsplitTotalinkm2mi21, each Text.Start(_, 3)))
```

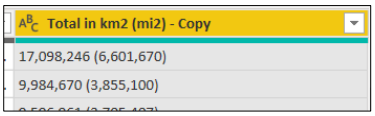
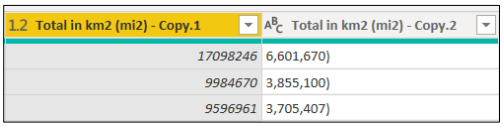
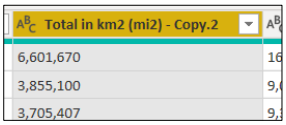
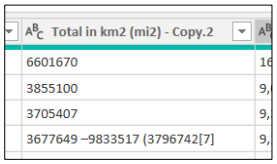
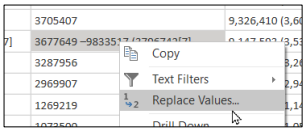
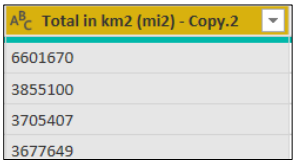
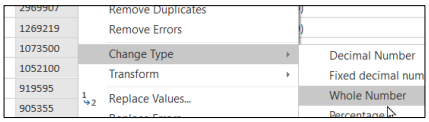
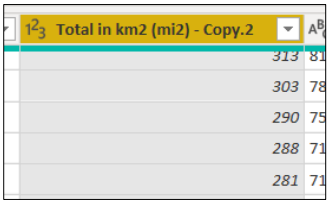
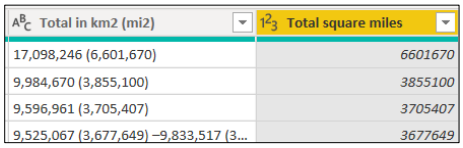
And here's what it does:

Stage	What it does	Returns for row 1
Before any formulae applied		**17,098,246 (6,601,670)**
splitTotalinkm2mi2 = Splitter.SplitTextByDelimiter(" (", QuoteStyle.None)([#"Total in km2 (mi2)"])	Creates a variable with a long unwieldy name (**splitTotalinkm2mi2**) to take the column called **Total in km2 (mi2)** and split it at the first (character. This will return a list of items.	A list containing two items: **17,098,246 (** and **6,601,670)**.
splitsplitTotalinkm2mi21 = Splitter.SplitTextByDelimiter(",", QuoteStyle.None)(splitTotalinkm2mi2{1}?)	Takes the second element returned in this list of items (list elements are numbered from 0, so [1] means the second element), which is the bit after the (character, and splits this by the , character.	A list containing the items **6**, **601** and **670)**.
Let … in Text.Combine(List.Transform(splitsplitTotalinkm2mi21, each Text.Start(_, 3)))	Takes this list of numbers, and for each one picks out the first 3 characters. Combines this list of miniature strings of text into a single string.	**6601670**

The above shows two things: that the M language won't be that intuitive to learn, and that (like all Microsoft wizards) the formulae created for columns by example are not always written in the simplest way!

2.3 Built-up Columns

An alternative to using column by examples is to create a new column one bit at a time. For our example (getting each country's total land area in square miles) this could be as follows:

Transform	How	Result
Duplicate column	Right-click on the total land area column and choose **Duplicate column** to take a copy of it, so we don't lose the original.	
Split by bracket	Right-click on this duplicated column and choose **Split Column** > **By Delimiter…,** then choose a (as your delimiter.	
Remove the end bracket	Right-click on the second column generated and choose **Replace values…**, then replace every) with nothing.	
Remove the commas	Repeat the above step, replacing each , with nothing.	
Remove the strange US reading	Right-click on the awkward US value cell, and choose to replace its value with 3677649: 	
Change the value to a number	Right-click on the column and choose to change the type to a number: 	
Tidy up the columns	Delete any intermediate columns generated and rename this one.	

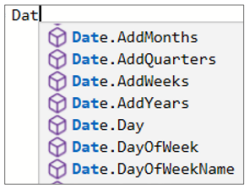
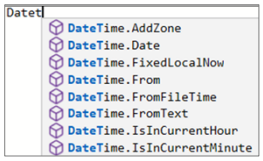
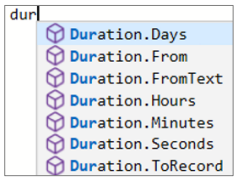
The great advantage of this approach is that you can understand how you got from A to B, reproduce the stages and even modify them should you require.

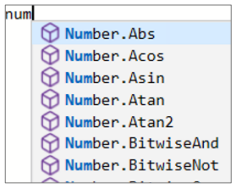
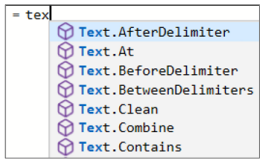
2.4 Custom Columns in M

Even if you don't know the M formula language you can sometimes guess your way round using it to create formulae for new columns based on existing ones.

M Prefixes

Here are some good prefixes to try in Intellisense:

Prefix	What it gives	Example
Date	Functions to do with dates	
DateTime	Functions to do with dates/times	
Duration	Functions giving the duration of time	

Prefix	What it gives	Example
Number	Functions giving things you can do to numbers	
Text	Functions to do with text	

Our Example – Elapsed Days

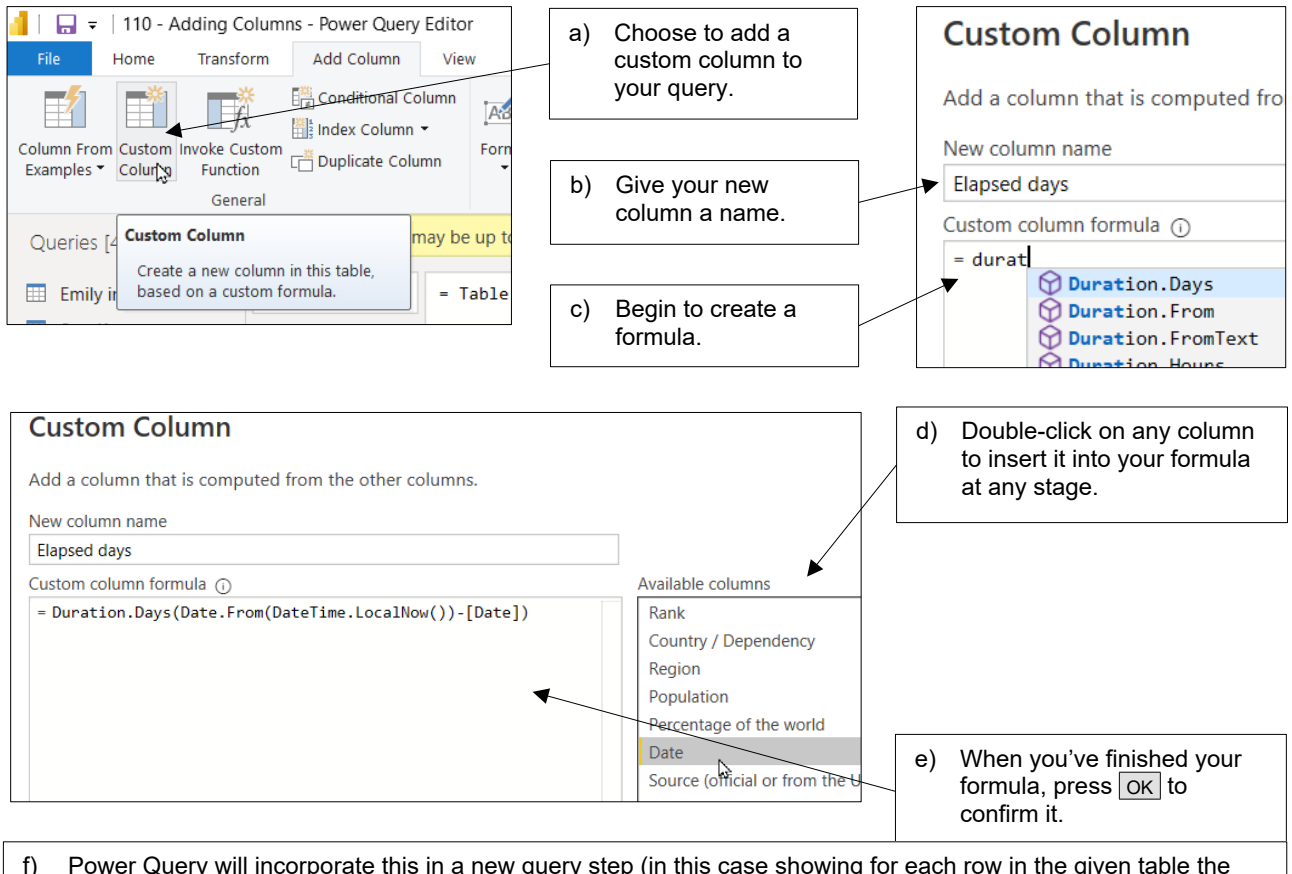
Suppose that you want to show the number of days which have elapsed since each country's population was last updated for our table

Date	Elapsed days
31/12/2021	94
03/04/2022	1
03/04/2022	1
01/07/2021	277
01/07/2021	277
03/04/2022	1

To understand how to create this formula, ask the question a different way: you want to take the duration in days of the period between the **[Date]** column in this table and the current local date/time, expressed as a date.

Creating a Custom Column

Here's how to create a column like this:



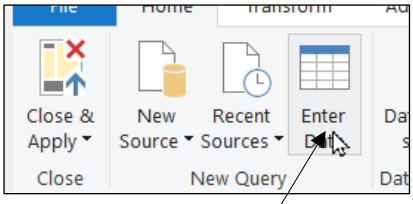
f) Power Query will incorporate this in a new query step (in this case showing for each row in the given table the number of days between the country's population estimate date and today's date/time, expressed as a date.

```
= Table.AddColumn(#"Filtered Rows1", "Elapsed days", each Duration.Days(Date.From(DateTime.LocalNow())-[Date]))
```

What the above example shows is that it's not easy to guess your way round the M language. Who would have thought it would be so hard to get at today's date, for example?

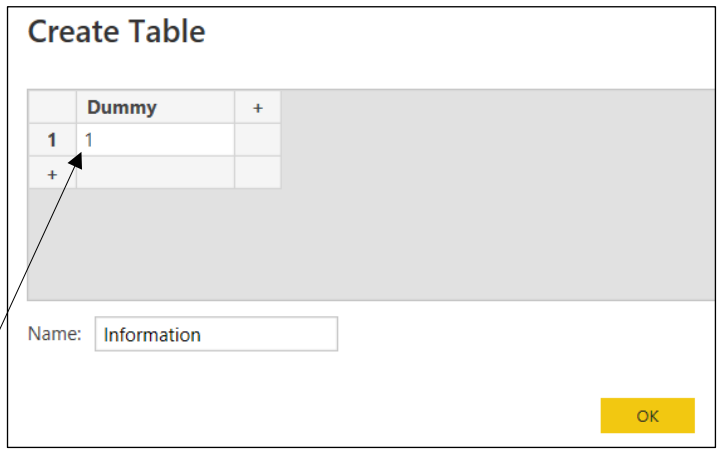
A Special Case: Last Refresh Date/time

This pages shows a way to show the last date/time when a report was refreshed:



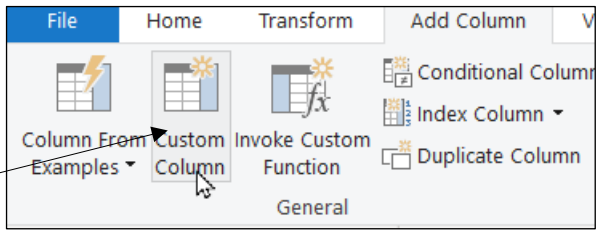
a) In Power Query, choose to create a new table by entering data for it.

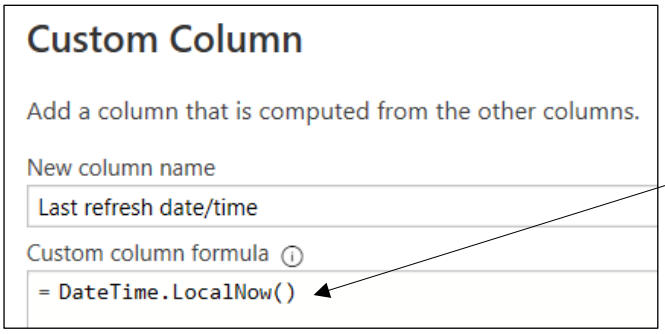
b) Make sure your table has one row in, since you want it to contain one bit of information only (it doesn't matter what this column is called, nor what it contains).



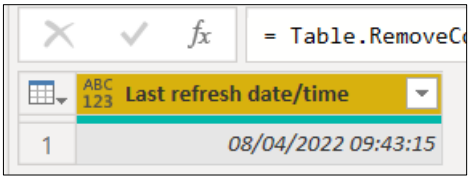
c) Select the table and column.

d) Choose to add a custom column from the **Add Column** tab of the ribbon.

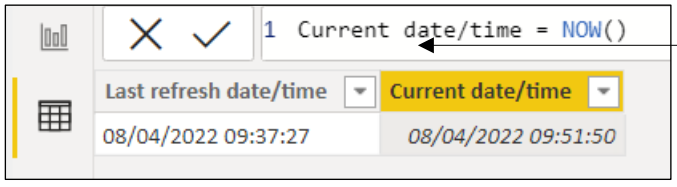




e) Create an expression giving the current date/time. You can then delete the original column to give your final table:

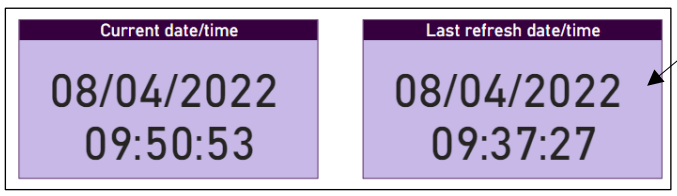


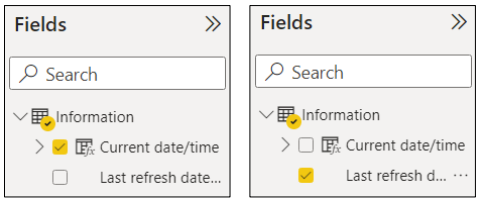
After loading your table, you could create a calculated column in it to show the current date and time, using the DAX **NOW()** function, then display both bits of information side by side:



You could add a column to your table in Power BI to show the latest date/time.

The card on the left is showing the date/time now; the one on the right is showing the date/time when you last refreshed your data:





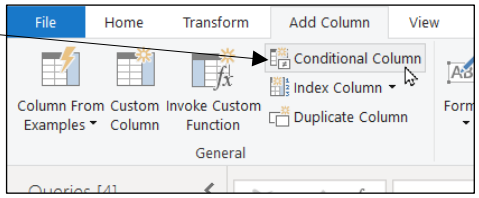
2.5 Conditional Columns

You can use conditional columns to divide data up into discrete bands. In the following example we express each country's land area in units of the size of Wales (about 8,023 square miles).

In an ideal world the provider of your data would add this column into the underlying data source, to prevent you having to calculate it as you load the data.

To create a conditional column like this:

a) On the **Add Column** tab select to create a conditional column.

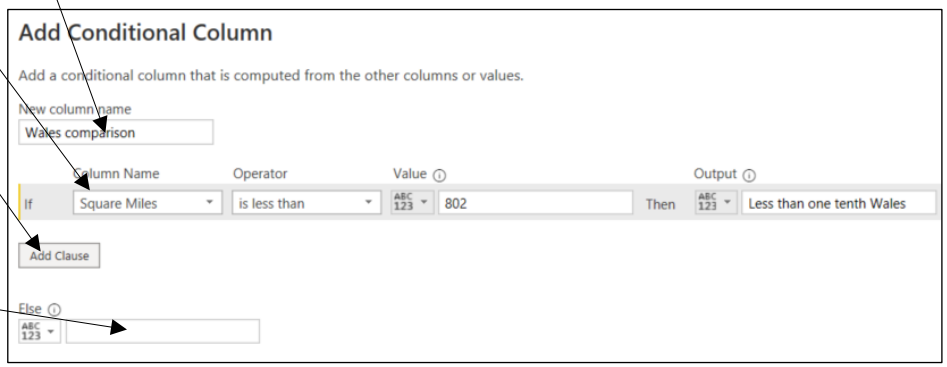


b) Give your new column a name (here we've called it **Wales comparison**).

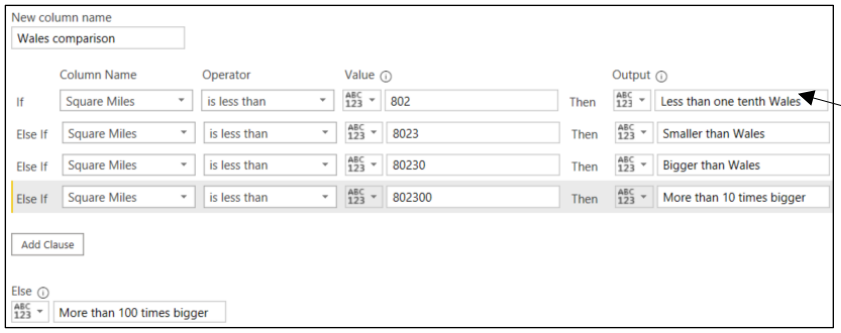
c) Create your first condition.

d) Click here to add other conditions.

e) Put here what should happen if all of the other conditions fail.



Continue adding conditions to get the final column:



Note that when writing numerical or date comparison conditions, you should start with the smallest/earliest number/date and work your way up (as here), or start with the biggest/latest one and work your way down.

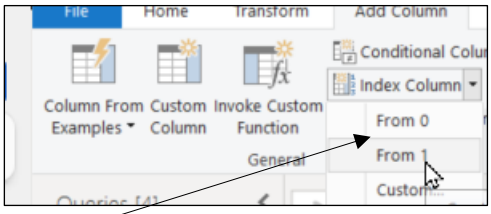
Country	Square Miles	Wales comparison
Afghanistan	252072	More than 10 times bigger
Albania	11100	Bigger than Wales
Algeria	919595	More than 100 times bigger
Andorra	181	Less than one tenth Wales
Angola	481400	More than 10 times bigger
Antigua and Barbuda	171	Less than one tenth Wales
Argentina	1073500	More than 100 times bigger

The first few countries listed in alphabetical order. The UK is more than 10 times bigger than Wales, if you're interested.

2.6 Indexing Columns

You can number rows in a table using an *index column*:

Rank	Country / Dependency	Custom
1	Russia	6601670
2	Canada[Note 3]	3855100
3/4	China	3705407
null	United States	3677649833517
5	Brazil	3287956
6	Australia	2969907



a) Sort your rows in the order in which you want them to be ordered (in this case, we'll use the original order that they were loaded in).

b) Choose to order the rows from 1, not 0. You can also choose **Custom…** to create any arithmetic series:

c) You can now use this column to show the original sort order (and to revert to it should you need to). In this example you could now delete the **Rank** column.

Index	Rank	Country / Dependency
1	1	Russia
2	2	Canada[Note 3]
3	3/4	China
4	*null*	United States
5	5	Brazil

What we do!

		Basic training	Advanced training	Systems / consultancy
Office	Microsoft Excel	✓	✓	✓
	VBA macros	✓	✓	✓
	Office Scripts	✓		
	Microsoft Access			✓
Power BI, etc	Power BI and DAX	✓	✓	✓
	Power Apps	✓		
	Power Automate (both)	✓	✓	
SQL Server	SQL	✓	✓	✓
	Reporting Services	✓	✓	✓
	Report Builder	✓	✓	✓
	Integration Services	✓	✓	✓
	Analysis Services	✓		
Coding	Visual C#	✓	✓	✓
	VB programming			✓
	MySQL	✓		
	Python	✓	✓	✓

Wise Owl Training


Wise Owl
Training