



Advanced Power BI (Data)

Sample manual - first two chapters



Wise Owl
Training

TABLE OF CONTENTS (1 of 5)

1	MANIPULATING COLUMNS	Page
1.1	Splitting Columns	7
	<i>Splitting by Number of Characters</i>	7
	<i>Splitting by Delimiters</i>	8
	<i>Splitting by Positions</i>	9
	<i>Splitting by Change in Case or Character Type</i>	9
	<i>Splitting into Rows</i>	10
	<i>Retaining Quotation Marks</i>	11
1.2	Merging Columns	12
1.3	Extracting Data	13
	<i>Replacing or Adding Columns</i>	13
	<i>An Example</i>	13
	<i>The Possible Options</i>	14

2	CREATING COLUMNS	Page
2.1	Ways to Create New Columns	15
2.2	Columns from Examples	16
	<i>Step 1 – Start the Feature</i>	16
	<i>Step 2 – Show some Examples</i>	17
	<i>Step 3 – Confirm the Formula</i>	17
	<i>Step 4 – Review your Formula</i>	17
	<i>Step 5 – Understanding your Formula</i>	18
2.3	Built-up Columns	19
2.4	Custom Columns in M	20
	<i>M Prefixes</i>	20
	<i>Our Example – Elapsed Days</i>	20
	<i>Creating a Custom Column</i>	21
	<i>A Special Case: Last Refresh Date/time</i>	22
2.5	Conditional Columns	23
2.6	Indexing Columns	24

3	COMBINING QUERIES	Page
3.1	Ways to Combine Data	25
3.2	Loading Multiple Files from a Folder	26
	<i>Setting the Folder</i>	26
	<i>Combining the Files</i>	27
	<i>Setting the Template for Import</i>	27
	<i>Tidying Up the Results</i>	28
	<i>An Outline of How it Works</i>	28
3.3	Appending Queries	29
	<i>Combining Queries using Append</i>	29
	<i>Tidying up your Results</i>	30
3.4	Merging Data	31
	<i>Loading the Tables to Merge</i>	31
	<i>Merging the Data</i>	32
	<i>Flattening the Combined Table</i>	33
	<i>Final Touches</i>	33
3.5	Fuzzy Merging	34
	<i>Setting a Similarity Threshold</i>	35
	<i>Enforce Case Sensitivity</i>	35
	<i>Using a Transformation Table</i>	35

4	MANIPULATION TRANSFORMS	Page
4.1	Unpivoting Data	36
	<i>Loading the Original Data</i>	36
	<i>Unpivoting the Data</i>	37
	<i>Renaming your Columns</i>	37
4.2	Grouping Data	38
	<i>Optional Step – Remove Excess Columns</i>	38
	<i>Starting Grouping</i>	39
	<i>Specifying How to Group Data</i>	39
4.3	Pivoting Columns	40
	<i>Remember Matrices</i>	40
	<i>Our Example for Pivot Columns</i>	40
	<i>Creating the Necessary New Columns</i>	41
	<i>Pivoting by Columns</i>	41
4.4	Transposing Data	42
	<i>Replacing within a Column</i>	42
	<i>Joining Rows Together</i>	43

TABLE OF CONTENTS (2 of 5)

5	NORMALISING DATA	Page
5.1	Our Example	44
5.2	Extracting Unique Lists	45
	<i>Duplicating the Original Table</i>	45
	<i>Creating a Unique List of Directors/Genres</i>	45
5.3	Numbering the Lists	46
5.4	Storing Id Numbers not Names	47
	<i>Merging the ID Column</i>	47
	<i>Expanding the Merged Columns</i>	48
	<i>Removing Unnecessary Additional Columns</i>	48
	<i>Making Sure the Right Table Loads</i>	48
	<i>The Final Relationships</i>	49
	<i>Finishing your Data Model</i>	49

6	DATA PROFILING	Page
6.1	Showing Profiling Data	50
	<i>Displaying Pop-up Column Profiles</i>	51
	<i>Grouping Profiled Data</i>	51
6.2	Query Dependencies	52

7	FILTERING QUERIES	Page
7.1	Removing Nulls and Errors	53
7.2	Filtering Dates	54
	<i>Previous or Next Time Periods</i>	54
	<i>Matching the Earliest/Latest Date</i>	55
	<i>Specific Time Periods</i>	55

8	ERROR HANDLING	Page
8.1	Default Treatment of Errors	56
	<i>Errors in the First 1000 Rows</i>	56
	<i>Errors in Subsequent Rows</i>	57
8.2	Fixing Errors in a Separate Table	58
	<i>Step 1 – Duplicate the Table</i>	58
	<i>Step 2 – Remove the Errors from One Table</i>	58
	<i>Step 3 – Correct the Errors in the Second Table</i>	59
	<i>Merging the Data Back Together</i>	59
8.3	Fixing Errors In Situ	60
	<i>Replacing Errors</i>	60
	<i>Handling with a Custom Column</i>	60

9	DATASETS AND DATAFLOWS	Page
9.1	Overview	61
9.2	Loading from a Power BI Dataset	62
9.3	Power BI Dataflows	63
	<i>Creating a Dataflow (the Data Source)</i>	63
	<i>Creating a Dataflow (the Transforms)</i>	64
	<i>Saving your Dataflow</i>	65
	<i>Loading Data from your Dataflow</i>	65

10	ROW-LEVEL SECURITY	Page
10.1	Overview of Row-Level Security	66
	<i>Limitations and Constraints</i>	66
10.2	Roles in Power BI Desktop	67
	<i>Creating Roles</i>	67
	<i>Testing Roles</i>	68
10.3	Roles in Power BI Service	69
	<i>Configuring the App Workspace</i>	69
	<i>Assigning Users to Roles</i>	70
	<i>Testing your Roles</i>	70
10.4	Dynamic Security	71
	<i>Creating a Permissions Table</i>	71
	<i>Creating a Role</i>	72
10.5	Creating Groups of Users	73
	<i>Creating the Tables</i>	73
	<i>Linking the Tables</i>	74
	<i>Creating the Role</i>	74

11	SWITCHING DATA SOURCES	Page
11.1	Changing the Data Source Manually	75
11.2	Switching the Data Source using M	76
	<i>Showing the Advanced Editor</i>	76
	<i>The Excel Query</i>	76
	<i>Creating a SQL Server Version of our Table</i>	77
	<i>Switching the Data Source</i>	77
	<i>Resolving Problems</i>	78
	<i>Viewing the Results</i>	78

TABLE OF CONTENTS (3 of 5)

12	SIMPLE PARAMETERS	Page
12.1	Parameters, Slicers and Filters	79
	<i>Parameters Needed for this Example</i>	79
12.2	Working with Parameters	80
	<i>Accessing Parameters</i>	80
	<i>Creating Parameters</i>	80
	<i>Managing Parameters</i>	81
	<i>Loading Parameters into Power BI</i>	81
12.3	Entering Values for Parameters	82
	<i>Entering Parameter Values in Query Editor</i>	82
	<i>Entering Parameter Values in Power BI Desktop</i>	82
12.4	Filtering using Parameters	83
12.5	Referencing Parameters in Power BI	84
	<i>Step 1 – Adding a Quick Measure to Reference the Parameter Value</i>	84
	<i>Step 2 – Create a Card to Display the Measure</i>	85
12.6	Using Parameters with Templates	86

13	DROPDOWN PARAMETERS	Page
13.1	Our Example	87
13.2	Creating a Simple List Parameter	88
13.3	Using List Queries for Parameters	89
	<i>Step 1 – Create a Query Giving a Single Column of Data</i>	89
	<i>Step 2 – Make this a List Query</i>	90
	<i>Step 3 – Creating the Dropdown Parameter</i>	90

14	PARAMETERS AND STORED PROCEDURES	Page
14.1	Overview	91
14.2	Loading Data from a Stored Procedure	92
14.3	Making Parameters Dynamic	93
	<i>Creating the Parameters</i>	93
	<i>Referencing the Parameters in the Advanced Editor</i>	94
	<i>Editing Parameter Values</i>	94

15	DYNAMIC CONNECTIONS	Page
15.1	SQL Server Dynamic Connections	95
	<i>First Enable Parameterisation of Data Sources</i>	95
	<i>Creating a Parameter</i>	96
	<i>Using this Parameter to Get Data</i>	96
	<i>Referencing your Parameter</i>	96
	<i>Changing the Server or Database</i>	97
	<i>The Query Editor Step Created</i>	97
15.2	Dynamic Connections to Excel	98
	<i>Step 1 – Create a Connection (Get Data)</i>	98
	<i>Step 2 – Create a Parameter</i>	99
	<i>Step 3 – Parameterising the Connection Used</i>	99
	<i>Step 4 - Changing the Worksheet Used</i>	100

16	WHAT-IF PARAMETERS	Page
16.1	Our Example	101
	<i>The Steps to Follow</i>	101
16.2	Creating and Using What-If Parameters	102
	<i>Step 1 – Create the What-If Parameter</i>	102
	<i>Step 2 – Customise the Slicer</i>	103
	<i>Step 3 – Create a Quick Measure to Show your Data</i>	104
	<i>Step 4 – Display your Measure in a Visual</i>	105

TABLE OF CONTENTS (4 of 5)

17	CUSTOM FUNCTIONS	Page
17.1	Overview	106
17.2	Example – Splitting Postcodes	107
	<i>Step 1 – Getting the Initial Data</i>	107
	<i>Step 2 - Creating a Parameter</i>	107
	<i>Step 3 - Duplicate your Parameter</i>	107
	<i>Step 4 – Create a Query Based on the Parameter Value</i>	108
	<i>Step 5 - Turning this One-Off Query into a Function</i>	108
	<i>Step 6 – Understanding the Resulting List of Queries</i>	109
	<i>Step 7 – Test your Function</i>	109
	<i>Step 8 – Invoke your Function for Each Table Row</i>	110
	<i>Step 9 – Expand the Results</i>	110
17.3	Example - Combining Selected Worksheets	111
	<i>Setting Up your Base Query and Parameter</i>	111
	<i>Link the Query to the Parameter</i>	112
	<i>Creating your Custom Function</i>	112
	<i>Invoking your Custom Function</i>	113
	<i>Expanding the Results</i>	113
17.4	Example – Combining All Worksheets	114
17.5	Combining Tables from Web Pages	115
	<i>Getting the Table of Genres</i>	115
	<i>Getting at the URL</i>	116
	<i>Creating a Parameter</i>	116
	<i>Getting the List of Films for a Genre</i>	116
	<i>Integrating the List of Films and your Parameter</i>	117
	<i>Creating the Function</i>	117
	<i>Invoking the Function</i>	118
17.6	Getting External Links	119

18	POWER APPS IN POWER BI	Page
18.1	Overview	120
	<i>Sharing Data Sources</i>	120
18.2	Embedding an App	121
	<i>Step 1 - Create a Power BI Report</i>	121
	<i>Step 2 – Adding an App Visual</i>	121
	<i>Step 3 – Creating or Choosing the App to Show</i>	122

19	POWER BI DESKTOP FLOWS	Page
19.1	Our Example	123
	<i>Some Limitations of Running Flows from Power BI Desktop</i>	123
19.2	Creating a Power Automate Button	124
	<i>Step 1 – Creating the Button</i>	124
	<i>Step 2 – Assigning Data Fields</i>	124
	<i>Step 3 – Start Editing your Flow</i>	125
	<i>Step 4 – Add Actions (and Data)</i>	125
	<i>Step 5 – Saving and Applying your Flow</i>	126
	<i>Step 6 – Formatting your Button</i>	126
	<i>Step 7 – Test your Button</i>	126
19.3	Editing Power BI Flows in Power Automate	127

20	USING APIS	Page
20.1	Introducing APIs	128
20.2	Some Simple Examples	129
	<i>Random Dog Pictures</i>	129
	<i>XKCD Cartoons</i>	130
	<i>Displaying Images Properly</i>	131
20.3	Case Study – Exchange Rates	132
	<i>Finding an API</i>	132
	<i>Getting the URL Needed</i>	132
	<i>Turning the Results into a Usable Table</i>	133
	<i>Adding Country Codes</i>	134
	<i>Getting the UK Exchange Rate</i>	134
20.4	Multiple Table APIs	135
	<i>Importing individual Tables</i>	135
	<i>Simplifying Column Titles</i>	136
	<i>The Need for a Bridging Table</i>	136
	<i>Starting the Bridging Table</i>	137
	<i>Removing Non-Essential Columns</i>	137
	<i>Expanding the List of Films</i>	137

21	HTML AND JSON	Page
21.1	Overview	138
21.2	Scraping HTML	139
	<i>Creating an HTML Table Function</i>	139
	<i>Expanding the Table Returned</i>	140
	<i>Getting the Readable Text Only</i>	140
21.3	Transforming JSON	141
	<i>Parsing JSON (or XML)</i>	141
	<i>Expanding the Resulting Record</i>	141
	<i>Extracting the List of Topics</i>	142
	<i>Creating the Relationship</i>	142



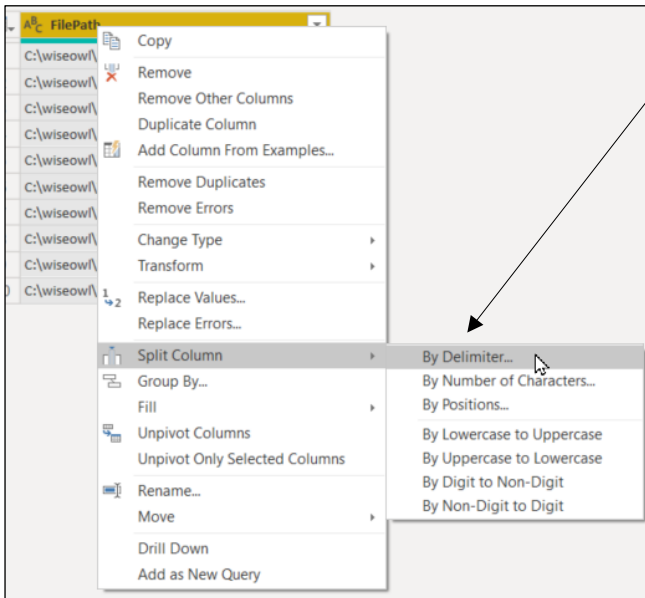
TABLE OF CONTENTS (5 of 5)

22	DATA INSIGHTS	Page
22.1	Overview of Data Insights	143
	<i>Three Warnings</i>	143
22.2	Text Analytics	144
	<i>Detecting Language</i>	144
	<i>Extracting Key Phrases</i>	145
	<i>Scoring Sentiment</i>	145
22.3	Tagging Images	146

CHAPTER 1 - MANIPULATING COLUMNS

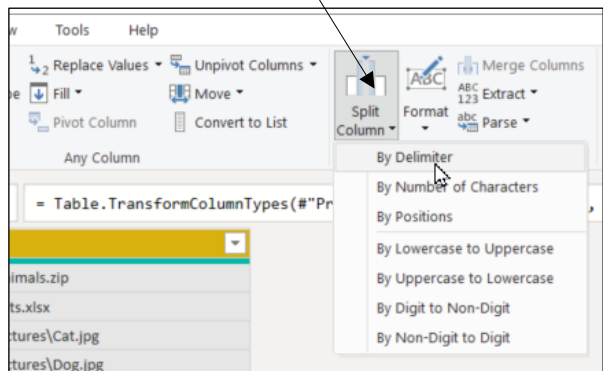
1.1 Splitting Columns

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

Split Column by Number of Characters

Specify the number of characters used to split the text column.

Number of characters

Split


Once, as far left as possible

Once, as far right as possible

Repeatedly

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:

AB C	FilePath
1	FilePath
2	C:\wiseowl\Text files\Small\To do list.txt
3	C:\wiseowl\Text files\Small\Shopping.txt
4	C:\wiseowl\Text files\Small\Reviewed.txt
5	C:\wiseowl\Text files\Small\Films.txt
6	C:\wiseowl\Text files\Small\Differences.txt
7	C:\wiseowl\Text files\Big\Big files.txt
8	C:\wiseowl\Pictures\Dog.jpg
9	C:\wiseowl\Pictures\Cat.jpg

a) Select the column you want to split by.

b) Choose a character to split by.

c) Choose the type of delimiter (see below for examples).

Split Column by Delimiter

Specify the delimiter used to split the text column.

Select or enter delimiter

--Custom--

\

Split at

Left-most delimiter

Right-most delimiter

Each occurrence of the delimiter

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

Option	Results	Notes																																								
<i>Left-most delimiter</i>	<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th>AB C</th> <th>FilePath.1</th> <th>AB C</th> <th>FilePath.2</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>C:</td> <td>wiseowl\Text files\Small</td> <td>To do list.txt</td> </tr> <tr> <td>2</td> <td>C:</td> <td>wiseowl\Text files\Small</td> <td>Shopping.txt</td> </tr> <tr> <td>3</td> <td>C:</td> <td>wiseowl\Text files\Small</td> <td>Reviewed.txt</td> </tr> </tbody> </table>	AB C	FilePath.1	AB C	FilePath.2	1	C:	wiseowl\Text files\Small	To do list.txt	2	C:	wiseowl\Text files\Small	Shopping.txt	3	C:	wiseowl\Text files\Small	Reviewed.txt	Split by the first backslash																								
AB C	FilePath.1	AB C	FilePath.2																																							
1	C:	wiseowl\Text files\Small	To do list.txt																																							
2	C:	wiseowl\Text files\Small	Shopping.txt																																							
3	C:	wiseowl\Text files\Small	Reviewed.txt																																							
<i>Right-most delimiter</i>	<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th>AB C</th> <th>FilePath.1</th> <th>AB C</th> <th>FilePath.2</th> </tr> </thead> <tbody> <tr> <td></td> <td>C:\wiseowl\Text files\Small</td> <td></td> <td>To do list.txt</td> </tr> <tr> <td></td> <td>C:\wiseowl\Text files\Small</td> <td></td> <td>Shopping.txt</td> </tr> <tr> <td></td> <td>C:\wiseowl\Text files\Small</td> <td></td> <td>Reviewed.txt</td> </tr> </tbody> </table>	AB C	FilePath.1	AB C	FilePath.2		C:\wiseowl\Text files\Small		To do list.txt		C:\wiseowl\Text files\Small		Shopping.txt		C:\wiseowl\Text files\Small		Reviewed.txt	Split by the last backslash																								
AB C	FilePath.1	AB C	FilePath.2																																							
	C:\wiseowl\Text files\Small		To do list.txt																																							
	C:\wiseowl\Text files\Small		Shopping.txt																																							
	C:\wiseowl\Text files\Small		Reviewed.txt																																							
<i>Each occurrence of the delimiter</i>	<table border="1" style="border-collapse: collapse; width: 100%;"> <thead> <tr> <th>AB C</th> <th>FilePath.1</th> <th>AB C</th> <th>FilePath.2</th> <th>AB C</th> <th>FilePath.3</th> <th>AB C</th> <th>FilePath.4</th> <th>AB C</th> <th>FilePath.5</th> </tr> </thead> <tbody> <tr> <td>C:</td> <td>wiseowl</td> <td></td> <td>Text files</td> <td></td> <td>Small</td> <td></td> <td>To do list.txt</td> <td></td> <td></td> </tr> <tr> <td>C:</td> <td>wiseowl</td> <td></td> <td>Text files</td> <td></td> <td>Small</td> <td></td> <td>Shopping.txt</td> <td></td> <td></td> </tr> <tr> <td>C:</td> <td>wiseowl</td> <td></td> <td>Text files</td> <td></td> <td>Small</td> <td></td> <td>Reviewed.txt</td> <td></td> <td></td> </tr> </tbody> </table>	AB C	FilePath.1	AB C	FilePath.2	AB C	FilePath.3	AB C	FilePath.4	AB C	FilePath.5	C:	wiseowl		Text files		Small		To do list.txt			C:	wiseowl		Text files		Small		Shopping.txt			C:	wiseowl		Text files		Small		Reviewed.txt			Split by every backslash
AB C	FilePath.1	AB C	FilePath.2	AB C	FilePath.3	AB C	FilePath.4	AB C	FilePath.5																																	
C:	wiseowl		Text files		Small		To do list.txt																																			
C:	wiseowl		Text files		Small		Shopping.txt																																			
C:	wiseowl		Text files		Small		Reviewed.txt																																			

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:

Split Column by Positions

Specify the positions at which to split the text column.

Positions

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:

The screenshot shows the 'Transform' menu in Power BI. The 'By Digit to Non-Digit' option is highlighted with a red box. Other options include 'By Delimiter', 'By Number of Characters', 'By Positions', 'By Lowercase to Uppercase', and 'By Uppercase to Lowercase'.

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!):

Split Column by Delimiter

Specify the delimiter used to split the text column.

Select or enter delimiter

--Custom--

-

Split at

Left-most delimiter

Right-most delimiter

Each occurrence of the delimiter

Advanced options

Split into

Columns

Rows

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

ABC 123	Product Code
	MPB-4242
	HQC-0233
	ZND-9682
	PGL-4142
	FFG-0033

ABC 123	Product Code	ABC 123	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

... 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).

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.

	A ^B _C Pet
1	"Annie","Cat",4,8.3
2	"Neo","Cat",4,8.5
3	"Tommy","Tortoise",4,6.0
4	"Pogba","Parrot",2,7.1

Here are the two options:

Option	Dialog box	Results for our example																				
Remove quote marks	<p>Split Column by Delimiter</p> <p>Specify the delimiter used to split the text column.</p> <p>Select or enter delimiter Comma</p> <p>Split at</p> <p><input type="radio"/> Left-most delimiter</p> <p><input type="radio"/> Right-most delimiter</p> <p><input checked="" type="radio"/> Each occurrence of the delimiter</p> <p>> Advanced options</p> <p>Quote Character "</p>	<table border="1"> <thead> <tr> <th></th> <th>A^B_C Pet.1</th> <th>A^B_C Pet.2</th> <th>A^B_C</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Annie</td> <td>Cat</td> <td>4</td> </tr> <tr> <td>2</td> <td>Neo</td> <td>Cat</td> <td>4</td> </tr> <tr> <td>3</td> <td>Tommy</td> <td>Tortoise</td> <td>4</td> </tr> <tr> <td>4</td> <td>Pogba</td> <td>Parrot</td> <td>2</td> </tr> </tbody> </table>		A ^B _C Pet.1	A ^B _C Pet.2	A ^B _C	1	Annie	Cat	4	2	Neo	Cat	4	3	Tommy	Tortoise	4	4	Pogba	Parrot	2
	A ^B _C Pet.1	A ^B _C Pet.2	A ^B _C																			
1	Annie	Cat	4																			
2	Neo	Cat	4																			
3	Tommy	Tortoise	4																			
4	Pogba	Parrot	2																			
Keep quote marks	<p>Split Column by Delimiter</p> <p>Specify the delimiter used to split the text column.</p> <p>Select or enter delimiter Comma</p> <p>Split at</p> <p><input type="radio"/> Left-most delimiter</p> <p><input type="radio"/> Right-most delimiter</p> <p><input checked="" type="radio"/> Each occurrence of the delimiter</p> <p>> Advanced options</p> <p>Quote Character None</p>	<table border="1"> <thead> <tr> <th></th> <th>A^B_C Pet.1</th> <th>A^B_C Pet.2</th> <th>A^B_C Pet</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>"Annie"</td> <td>"Cat"</td> <td>4</td> </tr> <tr> <td>2</td> <td>"Neo"</td> <td>"Cat"</td> <td>4</td> </tr> <tr> <td>3</td> <td>"Tommy"</td> <td>"Tortoise"</td> <td>4</td> </tr> <tr> <td>4</td> <td>"Pogba"</td> <td>"Parrot"</td> <td>2</td> </tr> </tbody> </table>		A ^B _C Pet.1	A ^B _C Pet.2	A ^B _C Pet	1	"Annie"	"Cat"	4	2	"Neo"	"Cat"	4	3	"Tommy"	"Tortoise"	4	4	"Pogba"	"Parrot"	2
	A ^B _C Pet.1	A ^B _C Pet.2	A ^B _C Pet																			
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:

	Product Code.1	Product Code.2	Sales
1	MPB	4242	
2	HQC	0233	
3	ZND	9682	
4	PGL	4142	
5	ESQ	9921	
6	SDB	9053	
7	CNA	4049	
8	UVU	0852	
9	CPU	4615	
10	PCS	2315	
11	YKF	3883	
12	TMZ	2076	

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).

Merge Columns

Choose how to merge the selected columns.

Separator:

New column name (optional):

	Original product code	Sales
1	MPB-4242	24.49703694
2	HQC-0233	44.9529665
3	ZND-9682	91.64723297
4	PGL-4142	30.60766405
5	ESQ-9921	29.87047969
6	SDB-9053	95.14955107

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

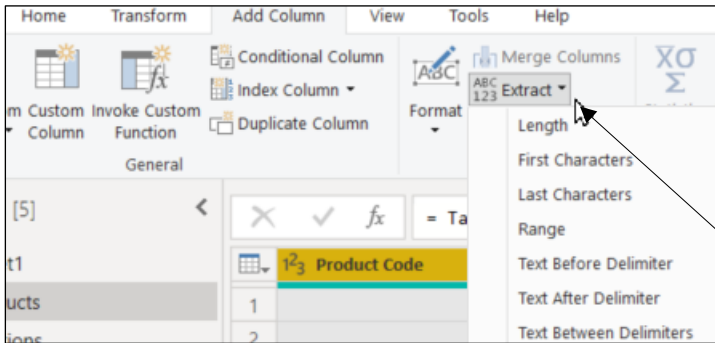
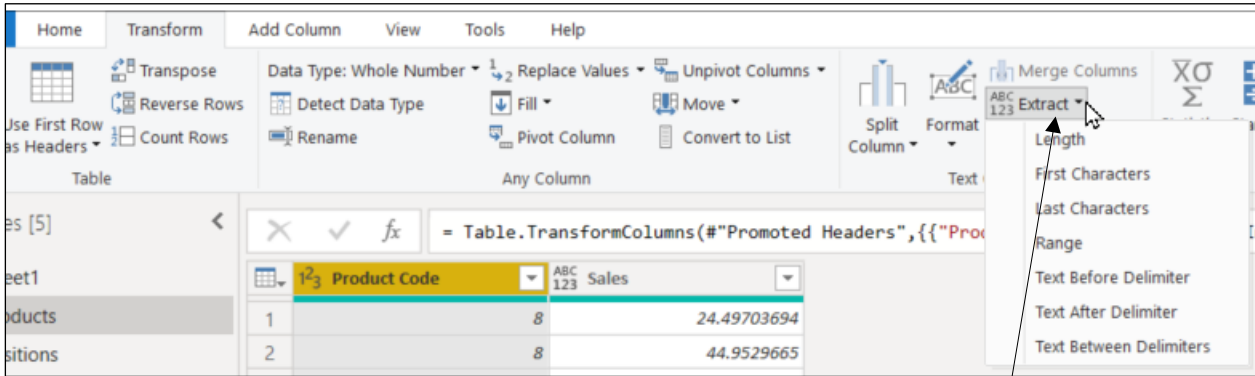


You're not limited to just two columns: you can merge as many columns as you like into a single one.

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.

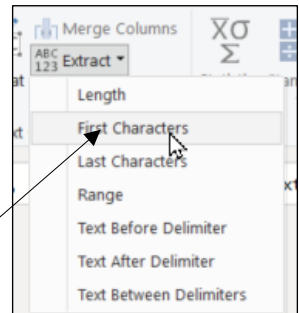
If you choose this version on the **Add Column** tab of the ribbon you will keep your existing column and add a new one to it.

An Example

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

	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

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).

Extract First Characters

Enter how many starting characters to keep.

Count

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

	Product code	Price
1	VPW	84.8176205
2	COS	94.1651118
3	MZZ	17.25046236
4	VWO	79.03115968

The Possible Options

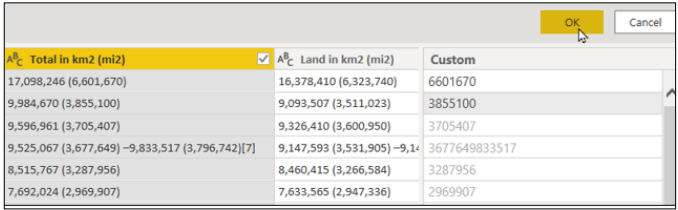
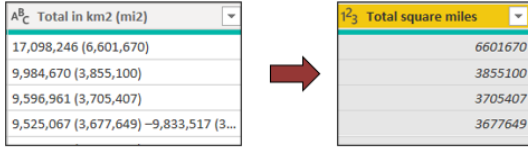
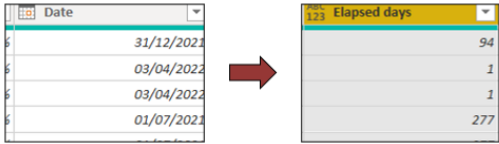
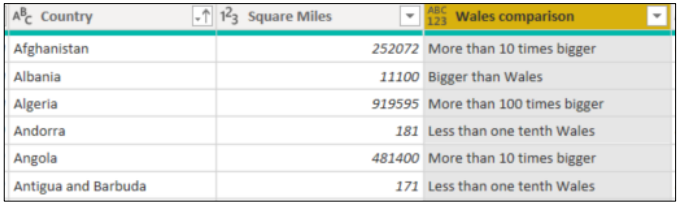

Here's what the options would give for a field containing the product code **VPW-85/L9-50**:

Option	Choices	Results	Notes
<i>Length</i>	None	12	This is a quick way to find how many characters there are for each value in a column.
<i>First Characters</i>	<div style="border: 1px solid black; padding: 5px;"> <p style="text-align: center;">Insert First Characters</p> <p>Enter how many starting characters to keep.</p> <p>Count 3</p> </div>	VPW	The (in this case) 3 left-most characters.
<i>Last Characters</i>	<div style="border: 1px solid black; padding: 5px;"> <p style="text-align: center;">Insert Last Characters</p> <p>Enter how many ending characters to keep.</p> <p>Count 2</p> </div>	50	The (in this case) 2 right-most characters.
<i>Range</i>	<div style="border: 1px solid black; padding: 5px;"> <p>Enter the index of the first character, and the number of characters to keep.</p> <p>Starting Index 4</p> <p>Number of Characters 2</p> </div>	85	The 2 characters starting at position number 4 (remember that characters are numbered from 0, not 1).
<i>Text Before Delimiter</i>	<div style="border: 1px solid black; padding: 5px;"> <p style="text-align: center;">Text Before Delimiter</p> <p>Enter the delimiter that marks the end of what you would like to extract.</p> <p>Delimiter A^BC -</p> <p>Advanced options Scan for the delimiter From the start of the input</p> <p>Number of delimiters to skip 1,2 0</p> </div>	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.
<i>Text After Delimiter</i>	<div style="border: 1px solid black; padding: 5px;"> <p style="text-align: center;">Text After Delimiter</p> <p>Enter the delimiter that marks the beginning of what you would like to extract.</p> <p>Delimiter A^BC -</p> <p>Advanced options Scan for the delimiter From the end of the input</p> <p>Number of delimiters to skip 1,2 0</p> </div>	50	
<i>Text Between Delimiters</i>	<div style="border: 1px solid black; padding: 5px;"> <p style="text-align: center;">Text Between Delimiters</p> <p>Enter the delimiters that mark the beginning and end of what you would like to extract.</p> <p>Start delimiter A^BC /</p> <p>End delimiter A^BC -</p> <p>Advanced options Scan for the start delimiter From the start of the input Number of start delimiters to skip 1,2 0 Scan for the end delimiter From the start delimiter, toward the... Number of end delimiters to skip 1,2 0</p> </div>	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
<i>Column from Examples</i>	Use AI to guess what formula you want to apply to all rows of a table based on one or two sample training values.	
<i>Built-up Columns</i>	Uses built-in Power Query transforms in sequence to produce complicated effects (you can then delete any intermediate columns created).	
<i>Custom Columns</i>	Returns the value - for each row of a table - of an expression using the M Power Query Formula language.	
<i>Conditional Columns</i>	Uses rules to divide rows up into a finite number of discrete categories.	
<i>Indexing Columns</i>	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

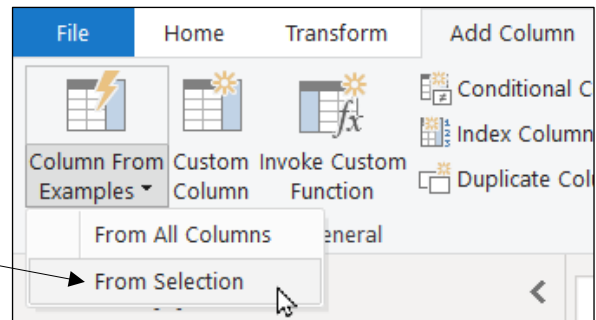
Begin by selecting the column or columns which contain the raw data from which Power Query can deduce the formula you want to apply, then right-click and choose this menu option:

Country / Dependency	Total in km2 (mi2)	Land in km2 (mi2)
Russia	17,098,246 (6,601,670)	
Canada[Note 3]	9,984,670 (3,855,100)	
China	9,596,961 (3,705,407)	
United States	9,525,067 (3,677,649)	-9,525,067
Brazil	8,515,767 (3,287,956)	
Australia	7,692,024 (2,969,907)	
India	3,287,263 (1,269,219)	
Argentina	2,780,400 (1,073,500)	
Kazakhstan	2,724,900 (1,052,100)	

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:

<input type="checkbox"/> A ^B _C Total in km2 (mi2)	<input checked="" type="checkbox"/> A ^B _C Land in km2 (mi2)	<input type="checkbox"/> A ^B _C 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** + **↵**.

<input checked="" type="checkbox"/> A ^B _C Total in km2 (mi2)	<input checked="" type="checkbox"/> A ^B _C 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) -9,147,593 (3,531,905)	

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).

Step 4 – Review your Formula

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

A ^B _C 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!

Custom Column

Add a column that is computed from the other columns.

New column name

Custom column 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)))
```

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:

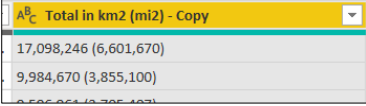
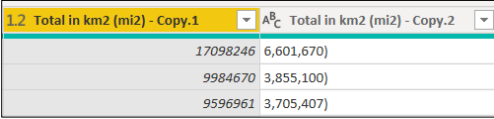
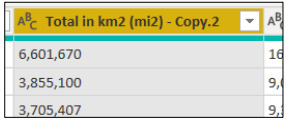
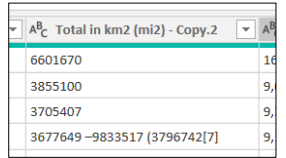
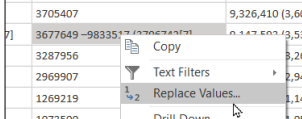

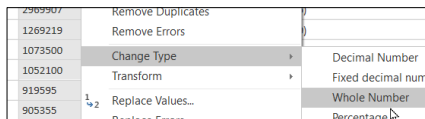
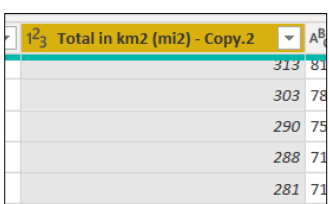
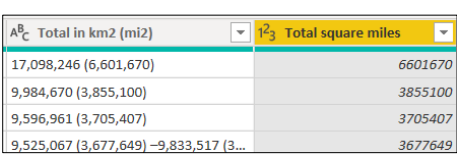
Stage	What it does	Returns for row 1
<i>Before any formulae applied</i>		17,098,246 (6,601,670)
<code>splitTotalinkm2mi2 = Splitter.SplitTextByDelimiter("(", QuoteStyle.None)([#"Total in km2 (mi2)"])</code>	Creates a variable with a long unwieldy name (splitTotalinkm2mi2) to take the column called Total in km2 (mi2) and split it at the first <code>(</code> character. This will return a list of items.	A list containing two items: 17,098,246 (and 6,601,670).
<code>splitsplitTotalinkm2mi21 = Splitter.SplitTextByDelimiter(",", QuoteStyle.None)(splitTotalinkm2mi2{1}?)</code>	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 <code>(</code> character, and splits this by the <code>,</code> character.	A list containing the items 6, 601 and 670).
<code>Let ... in Text.Combine(List.Transform(splitsplitTotalinkm2mi21, each Text.Start(_, 3)))</code>	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
<i>Duplicate column</i>	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.	
<i>Split by bracket</i>	Right-click on this duplicated column and choose Split Column > By Delimiter... , then choose a <code>[]</code> as your delimiter.	
<i>Remove the end bracket</i>	Right-click on the second column generated and choose Replace values... , then replace every <code>]]</code> with nothing.	
<i>Remove the commas</i>	Repeat the above step, replacing each <code>[]</code> with nothing.	
<i>Remove the strange US reading</i>	Right-click on the awkward US value cell, and choose to replace its value with 3677649: 	
<i>Change the value to a number</i>	Right-click on the column and choose to change the type to a number: 	
<i>Tidy up the columns</i>	Delete any intermediate columns generated and rename this one.	



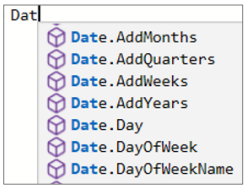
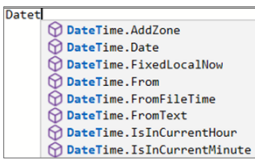
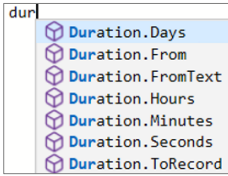
The great advantage of this approach is that you can understand how you got from A to B, reproduce the stages and even modify them should you require.

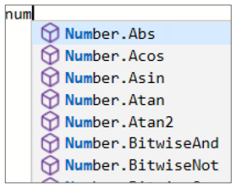
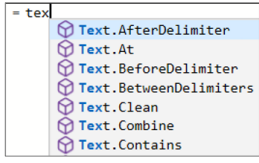
2.4 Custom Columns in M

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
<i>Date</i>	Functions to do with dates	
<i>DateTime</i>	Functions to do with dates/times	
<i>Duration</i>	Functions giving the duration of time	

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

Our Example – Elapsed Days

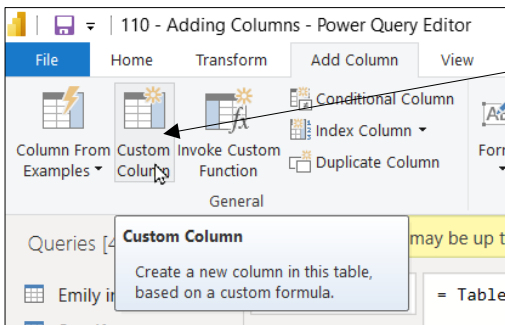
Suppose that you want to show the number of days which have elapsed since each country's population was last updated for our table

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
01/07/2021	277

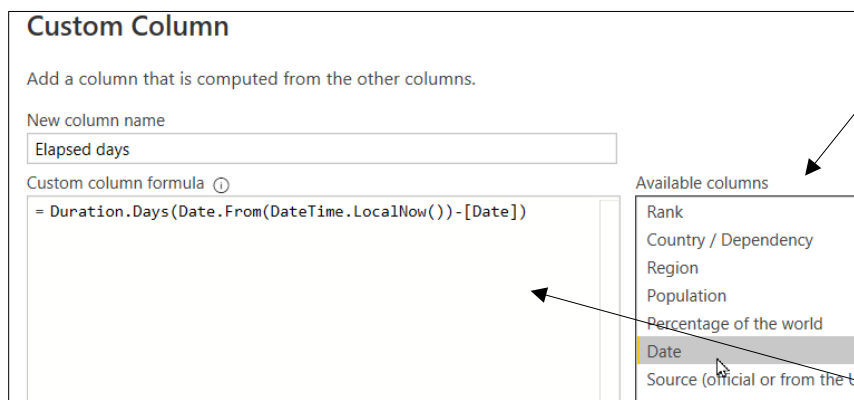
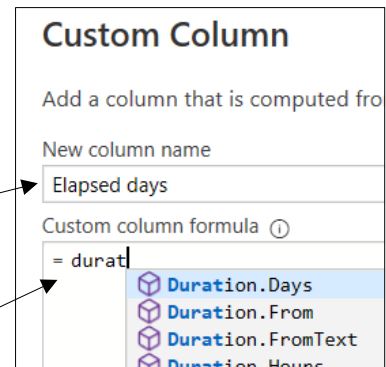
To understand how to create this formula, ask the question a different way: you want to take the duration in days of the period between the **[Date]** column in this table and the current local date/time, expressed as a date.

Creating a Custom Column

Here's how to create a column like this:




- a) Choose to add a custom column to your query.
- b) Give your new column a name.
- c) Begin to create a formula.



- d) Double-click on any column to insert it into your formula at any stage.
- e) When you've finished your formula, press **OK** to confirm it.

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]))
```

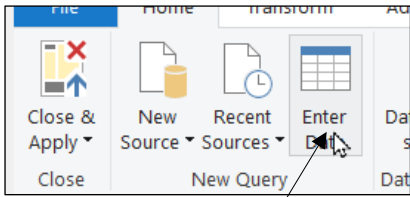


Wise Owl's Hint

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?

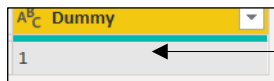
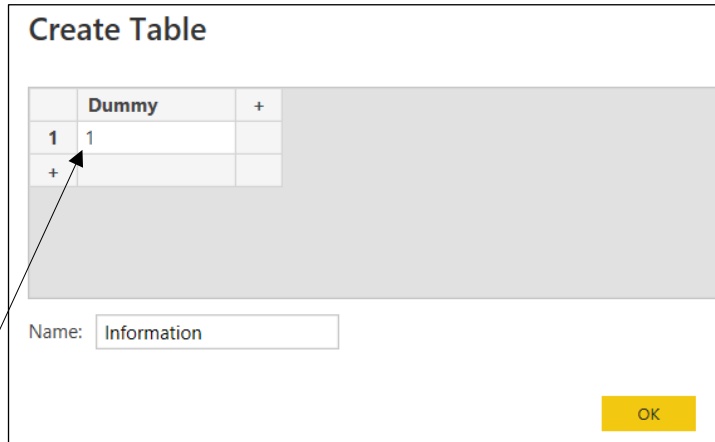
A Special Case: Last Refresh Date/time

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



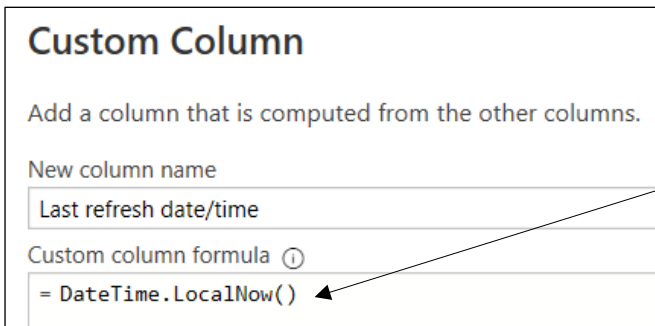
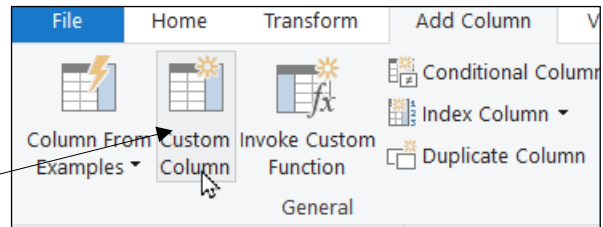
a) In Power Query, choose to create a new table by entering data for it.

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).

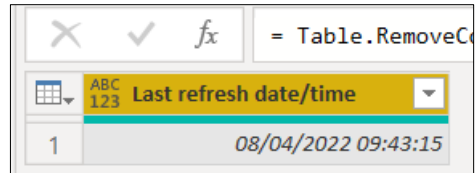


c) Select the table and column.

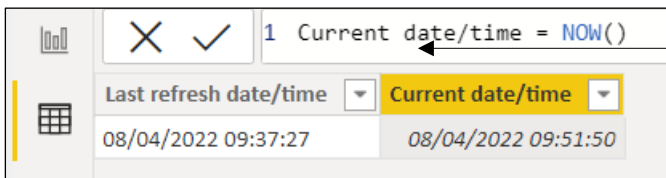
d) Choose to add a custom column from the **Add Column** tab of the ribbon.



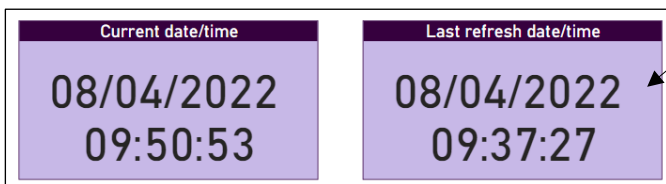
e) Create an expression giving the current date/time. You can then delete the original column to give your final table:



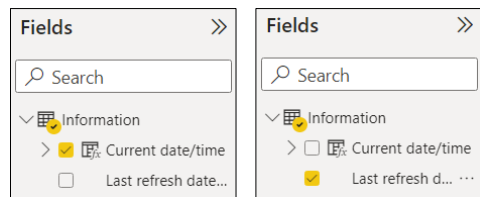
After loading your table, you could create a calculated column in it to show the current date and time, using the DAX **NOW()** function, then display both bits of information side by side:



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.

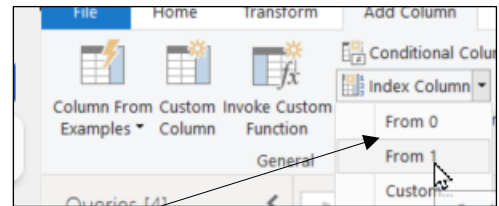
Country	Square Miles	Wales comparison
Afghanistan	252072	More than 10 times bigger
Albania	11100	Bigger than Wales
Algeria	919595	More than 100 times bigger
Andorra	181	Less than one tenth Wales
Angola	481400	More than 10 times bigger
Antigua and Barbuda	171	Less than one tenth Wales
Argentina	1073500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

Add Index Column

Add an index column with a specified starting index and increment.

































Starting Index

Increment

- 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		<i>null</i> 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	