



Happy New Year! After two horrendous years, let's hope 2022 starts to see economic turnarounds and COVID-19 well and truly a thing of the past. Fingers crossed for the upcoming year!

We start 2022 with improvements to Excel for the Web and a rant on good ol' **VLOOKUP**. No hangovers here: the regulars are all present and correct too: we have another Beat the Boredom Challenge, Charts & Dashboards, Visual Basics, Power Pivot Principles, Power Query Pointers and the usual plethora of Power BI updates. We consult the **HYP**sters on the A to Z of Excel Functions and the Keyboard Shortcuts take **CTRL** later in the piece.

Good riddance 2021!

As always, happy reading and remember: stay safe, stay happy, stay healthy.

Liam Bastick, Managing Director, SumProduct



New Features in Excel for the Web

Microsoft decided to get busy before they broke up for the seasonal year-end break. Excel for the web benefitted immensely with a raft of new features / updates:

- searchable data validation list
- Queries pane
- text in shapes
- paste option floaty
- sheet direction switcher
- open and edit more files.

Searchable Data Validation List

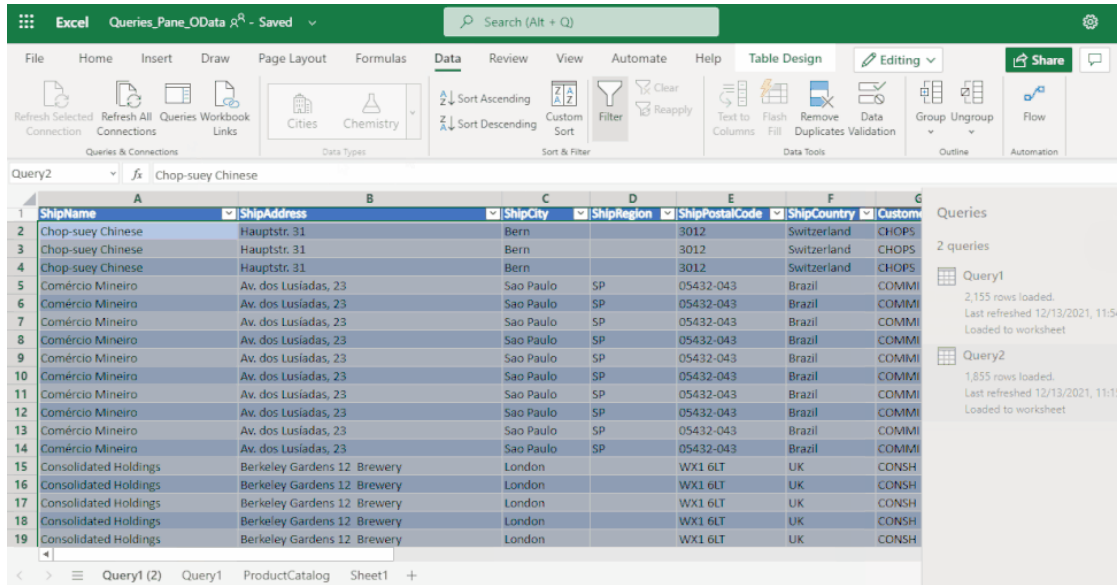
Selecting an item from a drop-down list is now simpler, especially when the list is long. You may search within the data validated list by typing any letter that is part of the item you're looking for and Excel for the web will

help auto-complete the search for you. Blank values are hidden from the drop-down list and duplicate values are removed.

Tech professions	DV dropdown list
Analyst	man
Data scientist	IT manager
Database administrator	Product manager
Information security analyst	Program manager
Information systems engineer	
IT manager	
Product manager	
Program manager	
Researcher	
Sales engineer	
Software developer	
Software engineer	
Support engineer	
UX/UI designer	
Web developer	

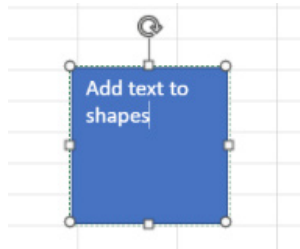
Queries Pane

The full Power Query experience is rolling out to Excel for the web. Coming first is the Queries pane, with the familiar user experience of the 'Queries & Connections' pane in Excel Desktop. With this Queries pane, you can view all the queries in the workbook in Excel for the web.

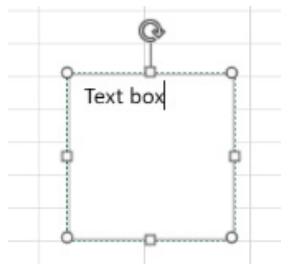


Text in Shapes

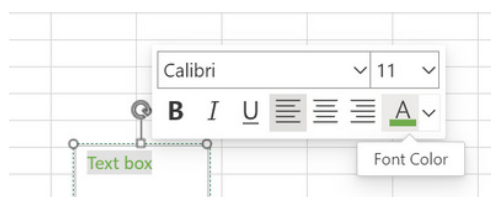
If you're using shapes to add design and context to your data and analysis, you should note that you may now add or edit text in shapes. Simply select a shape and start typing to add or edit text in the shape. Alternatively, double-click on the shape or right click and select 'Edit text'.



Furthermore, you may now add a text box by selecting the **Insert Ribbon tab -> Shapes -> Text box** and start typing to add text.

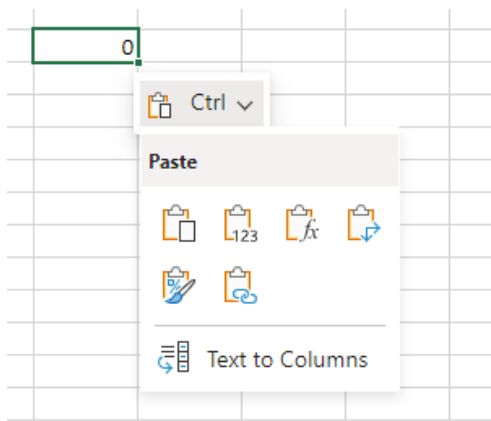


The text may be formatted too. You can change font style, size and colour using the text formatting options on the Home Ribbon tab or via the quick option floaty when you select and right click on the text.



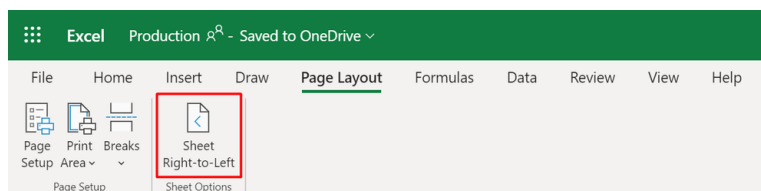
Paste Option Floaty

You may also now access all the Paste options right where you're pasting content or after:



Sheet Direction Switcher

For users preferring the right-to-left (RTL) experience, you may now flip a spreadsheet to RTL at the click of a button on the 'Page Layout' Ribbon tab. Previously available only in Excel Desktop, switching to RTL – and back – is now also now available in Excel for the web.

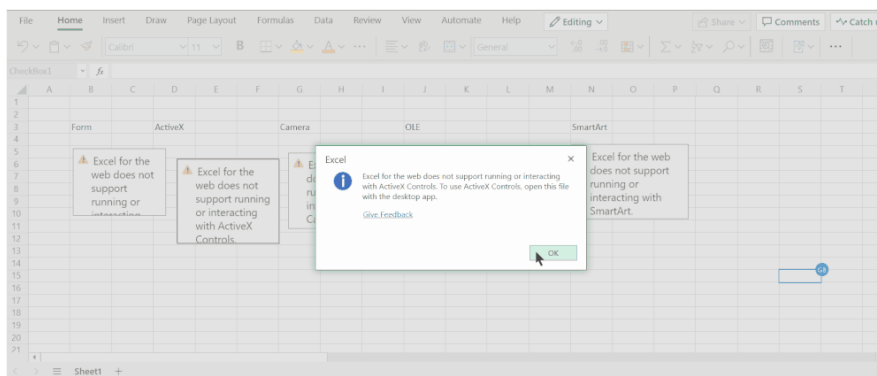


Open and Edit More Files

As can be seen from above, Excel for the web continues to add features and improvements to close the gap with Desktop. To this end, you may now open and edit more files on the web and complete your job anywhere, for any given browser.

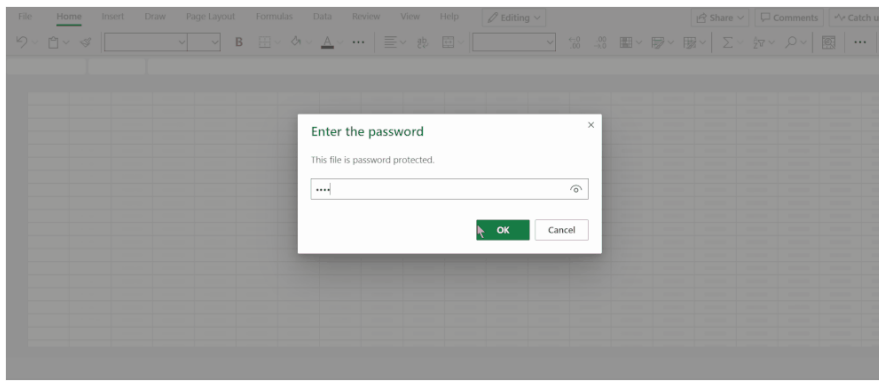
For instance, you may now open and edit large files from SharePoint up to 100MB. This allows you to work with large numbers of rows of data and complete your work without having to switch to the Desktop.

Furthermore, you may also now edit files containing Legacy Art objects and SmartArt objects in Excel for the Web. These legacy art objects include Form Controls, ActiveX Controls, Camera Tools and OLE objects. This enables you to access and interact with any of such workbooks, but not with the objects themselves. If you are using either a Button form control or an ActiveX control to run a macro, watch out for the upcoming Office Scripts functionality that will allow you to run Office Scripts with a workbook button.



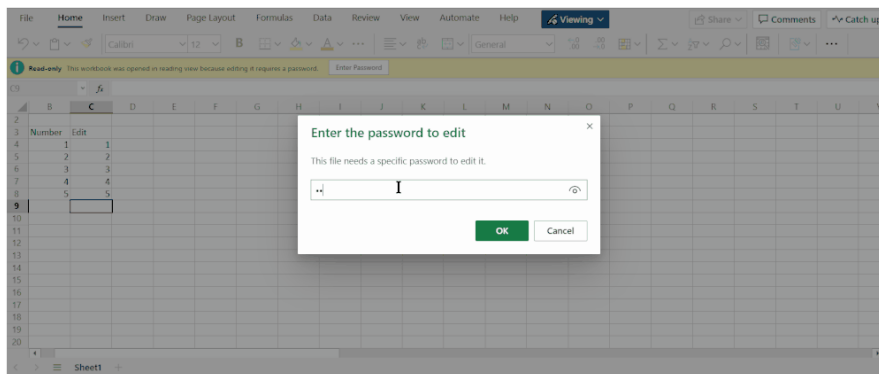
You can also now access your password protected workbooks on Excel for the Web. This allows you to open and interact with your files protected with a password to open without switching back to Desktop. However, it should be noted that co-authoring is not supported with password

protected files (which is the same as with Excel Desktop). In this scenario, Microsoft recommends you switch to the new Microsoft Information Protection (MIP) encrypted workbooks that allow you to co-author and still protect sensitive data.



But that's not all. You may also edit files which are protected for editing using a password, thus allowing you to access / interact with different types of protected workbooks on the browser itself. As noted

above, co-authoring is not supported for Password to Modify (again, same as Desktop), therefore, here it's recommended that you share edit access to specific people that allows you to block editing to others.



More is to come. The following capabilities will soon be coming to Excel for the Web:

- support for large files up to 100 MB from OneDrive (only for M365 subscribers) in Excel for the Web, thus enabling you to work with large datasets in the browser
- editing of files with the presence of legacy shared workbook feature, by provisioning you with one-click turn-off of the legacy feature, thus allowing you to interact and collaborate with the workbook
- editing of files containing the legacy data wizard connections like Text Queries, Web Queries, Query Table or SharePoint Lists, thus enabling interaction with the workbook. Do note that Microsoft has stated that they do not plan to support the connection itself and recommend you should switch to import data via Power Query to connect and refresh data.

Potential New Year's Resolution?

With 2022 now upon us, it's not too late to make a New Year's Resolution. Never mind giving up something for Lent, how about giving something up because of the associated issues? I thought I'd start off the New Year with a **VLOOKUP** rant!!

Often we need to look up data in a table or list – and two functions most of us are very familiar with are **VLOOKUP** and **HLOOKUP**. But do you realise it's very easy to make a mistake with these functions?

Let's start with a refresher.

VLOOKUP(lookup_value, table_array, column_index_number, [range_lookup]) has the following syntax:

- **lookup_value**: what value do you want to look up?
- **table_array**: where is the lookup table?
- **column_index_number**: which column has the value you want returned?
- **[range_lookup]**: do you want an exact or an approximate match? This is optional and to begin with, I am going to ignore this argument exists.

HLOOKUP is similar, but works on a row, rather than a column, basis.

Example

I am going to use **VLOOKUP** throughout to keep things simple. **VLOOKUP** always looks for the **lookup_value** in the first column of a table (the **table_array**) and then returns a corresponding value so many columns to the right, determined by the **column_index_number**.

G25 : X ✓ fx =VLOOKUP(G23,F13:M18,8)

Lookup Value	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Result Column
1	94	82	36	58	99	32	84
2	40	77	75	28	96	48	47
3	92	55	96	57	50	81	80
4	61	47	45	51	27	39	35
5	11	52	67	86	96	11	66
6	76	96	29	84	45	26	44

VLOOKUP

Value: 2

Result: 47

In this above example, the formula in cell G25 seeks the value 2 in the first column of the table F13:M18 and returns the corresponding value from the eighth column of the table (returning 47).

Pretty easy to understand; so far so good. So what goes wrong? Well, what happens if you add or remove a column from the table range?

Adding (inserting) a column gives us the wrong value:

G25 : X ✓ fx =VLOOKUP(G23,F13:N18,8)

Lookup Value	Column 2	Column 3	Column 4	Column 5	Column 6	INSERTED	Column 7	Result Column
1	94	82	36	58	99		32	84
2	40	77	75	28	96		48	47
3	92	55	96	57	50		81	80
4	61	47	45	51	27		39	35
5	11	52	67	86	96		11	66
6	76	96	29	84	45		26	44

VLOOKUP

Value: 2

Result: 48

With a column inserted, the formula contains hard code (8) and therefore, the eighth column (M) is still referenced, giving rise to the wrong value. Deleting a column instead is even worse:

G25 : X ✓ fx =VLOOKUP(G23,F13:L18,8)

Lookup Value	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Result Column
1	94	82	36	58	99	32	84
2	40	77	75	28	96	48	47
3	92	55	96	57	50	81	80
4	61	47	45	51	27	39	35
5	11	52	67	86	96	11	66
6	76	96	29	84	45	26	44

VLOOKUP

Value: 2

Result: #REF!

Now there are only seven columns so the formula returns #REF! Oops.

It is possible to make the column index number dynamic using the COLUMNS function:

G25 : X ✓ fx =VLOOKUP(G23,F13:M18,COLUMNS(F13:M13))

Lookup Value	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Result Column
1	94	82	36	58	99	32	84
2	40	77	75	28	96	48	47
3	92	55	96	57	50	81	80
4	61	47	45	51	27	39	35
5	11	52	67	86	96	11	66
6	76	96	29	84	45	26	44

VLOOKUP

Value: 2

Result: 47

COLUMNS(reference) counts the number of columns in the **reference**. Using the range **F13:M13**, this formula will now keep track of how many columns there are between the lookup column (**F**) and the result column (**M**). This will prevent the problems illustrated above.

But there are more issues. Consider duplicate values in the lookup column. With one duplicate, the following happens:

G25 :

Lookup Value	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Result Column
2	94	82	36	58	99	32	84
2	40	77	75	28	96	48	47
3	92	55	96	57	50	81	80
4	61	47	45	51	27	39	35
5	11	52	67	86	96	11	66
6	76	96	29	84	45	26	44

VLOOKUP

Value:

Result:

Here, the second value is returned, which might not be what is wanted. With two duplicates:

G25 :

Lookup Value	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Result Column
2	94	82	36	58	99	32	84
2	40	77	75	28	96	48	47
2	92	55	96	57	50	81	80
4	61	47	45	51	27	39	35
5	11	52	67	86	96	11	66
6	76	96	29	84	45	26	44

VLOOKUP

Value:

Result:

Ah, it looks like it might take the last occurrence. Testing this hypothesis with three duplicates:

G25 :

Lookup Value	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Result Column
2	94	82	36	58	99	32	84
2	40	77	75	28	96	48	47
2	92	55	96	57	50	81	80
2	61	47	45	51	27	39	35
5	11	52	67	86	96	11	66
6	76	96	29	84	45	26	44

VLOOKUP

Value:

Result:

Yes, there seems to be a pattern: **VLOOKUP** takes the last occurrence. Better make sure:

G25 :

Lookup Value	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Result Column
2	94	82	36	58	99	32	84
2	40	77	75	28	96	48	47
2	92	55	96	57	50	81	80
2	61	47	45	51	27	39	35
5	11	52	67	86	96	11	66
2	76	96	29	84	45	26	44

VLOOKUP

Value:

Result:

Rats. In this example, the value returned is the fourth of five. The problem is, there's no consistent logic and the formula and its result cannot be relied upon. It gets worse if we exclude duplicates but mix up the lookup column a little:

G25 : $\text{=VLOOKUP}(G23,F13:M18,COLUMNS(F13:M13))$

Lookup Value	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Result Column
5	94	82	36	58	99	32	84
2	40	77	75	28	96	48	47
3	92	55	96	57	50	81	80
(4)	61	47	45	51	27	39	35
1	11	52	67	86	96	11	66
5	76	96	29	84	45	26	44

VLOOKUP

Value: 2

Result: #N/A

In this instance, **VLOOKUP** cannot even find the value 2!

So what's going on? The problem – and common modelling mistake – is that the fourth argument has been ignored:

VLOOKUP(lookup_value, table_array, col_index_number, [range_lookup])

[range_lookup] appears in square brackets, which means it is optional. It has two values:

- **TRUE:** this is the default setting if the argument is not specified. Here, **VLOOKUP** will seek an approximate match, looking for the largest value less than or equal to the value sought. There is a price to be paid though: the values in the first column (or row for **HLOOKUP**) must be in strict ascending order – this means that each value must be larger than the value before, so no duplicates.

This is useful when looking up postage rates for example where prices are given in categories of pounds and you have 2.7lb to post (say). It's worth noting though that this isn't the most common lookup when modelling.

- **FALSE:** this has to be specified. In this case, data can be any which way – including duplicates – and the result will be based upon the first occurrence of the value sought. If an exact match cannot be found, **VLOOKUP** will return the value #N/A.

And this is the problem highlighted by the above examples. The final argument was never specified so the lookup column data has to be in strict ascending order – and this premiss was continually breached.

The robust formula needs both **COLUMNS** and a fourth argument of **FALSE** to work as expected:

G25 : $\text{=VLOOKUP}(G23,F13:M18,COLUMNS(F13:M13),FALSE)$

Lookup Value	Column 2	Column 3	Column 4	Column 5	Column 6	Column 7	Result Column
6	94	82	36	58	99	32	84
2	40	77	75	28	96	48	47
#DIV/0!	92	55	96	57	50	81	80
2	61	47	45	51	27	39	35
2	11	52	67	86	96	11	66
(5)	76	96	29	84	45	26	44

VLOOKUP

Value: 2

Result: 47

This is a very common mistake in modelling. Using a fourth argument of **FALSE**, **VLOOKUP** will return the corresponding result for the first occurrence of the **lookup_value**, regardless of number of duplicates, errors or series order. If an approximate match is required, the data must be in strict ascending order.

VLOOKUP (and consequently **HLOOKUP**) are not the simple, easy to use functions people think they are. In fact, they can never be used to return data for columns to the left (**VLOOKUP**) or rows above (**HLOOKUP**). Perhaps **INDEX MATCH**, **LOOKUP** and / or **XLOOKUP** are more viable options!

Word to the Wise

As stated above, **HLOOKUP** works like **VLOOKUP** but hunts out a value in the first row of a table and returns a value so many rows below this reference. However, it has the same limitations and should be used just as carefully.

Beat the Boredom Challenge

With many of us currently “working from home” / quarantined, there are only so Zoom / Teams calls and virtual parties you can make before you reach your (data) limit. Perhaps they should measure data allowance in blood pressure millimetres of mercury (mmHg). To try and keep our

readers engaged, we will continue to reproduce some of our popular **Final Friday Fix** challenges from yesteryear in this and upcoming newsletters. One suggested solution may be found later in this newsletter. Here’s this month’s...

Dates are not quite what they seem in Excel. They can cause issues for the unwary, so they are important to understand. For example, August 17, 2008 may be expressed as:

- 8/17/08 (month/day/year, US format)
- 17-8-08 (day – month - year, UK / Europe format, as it’s called by Microsoft, but pretty much anywhere else)
- August 17, 2008
- 17 Aug 2008
- ...

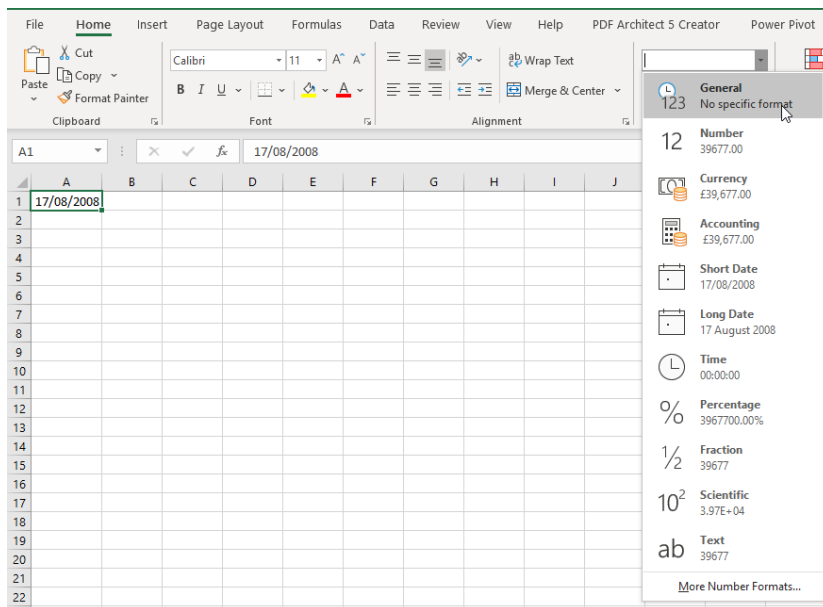
Therefore, it does not make sense for Excel to try and force a particular format and many formats are thus supported. Indeed, one thing that is more sophisticated is the fact that whether by default, day comes before month or vice versa is down to your regional settings. This means if I type in

17/8/08

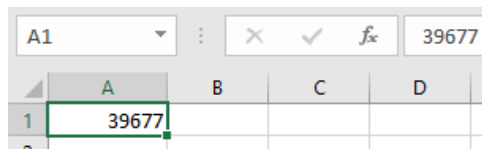
on some computers, this will be recognised as a date, whereas on others it will be seen as a ‘General’ text format because 17 exceeds the number

of months in a year. For our American readers, I apologise in advance: my computer is not set to US settings and therefore, the examples here will be displayed as day followed by month followed by year. In the US, the month preceded the day.

The point is, these are display issues only. As a non-US Excel user, I will type “17/8/08” into cell **A1**, and then format the cell as ‘General’ (using the drop-down box in the ‘Number’ grouping of the ‘Home’ tab):



This changes cell **A1** as follows:



It becomes “39677”, which is known as a **serial number** (if I were to delete this value, it would make me a serial killer). This serial number is a count from “Day 1” which is defined as 1 January 1900, so Day 2 would be 2 January 1900, Day 3 would be 3 January 1900, and so on. Therefore, according to Excel, 17 August 2008 is Day 39,677.

In essence, we just need Excel to recognise the date as a date – and that’s this month’s challenge...

The Challenge

This month’s challenge is to convert from US date format (month – day – year) to what is known as the non-US, or more commonly, the European, date format (day – month – year). If you’re in the US, the challenge is the opposite!

The problem arises when you receive date data in a spreadsheet that is not recognised by your regional settings. Therefore, for me, my computer cannot make sense of US date formats such as

	A
1	Dates
2	8/13/20
3	6/14/21
4	12/05/2022
5	9/30/23
6	1/29/24
7	8/17/25
8	5/22/26
9	3/21/27
10	7/14/28
11	5/25/29
12	

I have left the data in 'General' style deliberately so you can see only one entry, cell **A4**, is recognised as a number (date). The problem is, even that's wrong as that represents 5 December 2022 not 12 May 2022.

How do I convert it? The simplest answer would be to use Power Query / Get & Transform – but that's not what I want here. It may also be done with Excel's 'Text to Columns' feature, but that's not the solution here either. I want one formula to do the following:

	A	B	C
1	Dates		Converted
2	8/13/20		13/08/2020
3	6/14/21		14/06/2021
4	12/05/2022		05/12/2022
5	9/30/23		30/09/2023
6	1/29/24		29/01/2024
7	8/17/25		17/08/2025
8	5/22/26		22/05/2026
9	3/21/27		21/03/2027
10	7/14/28		14/07/2028
11	5/25/29		25/05/2029
12			

Sound easy? Try it. One solution just might be found later in this newsletter – but no reading ahead!

Charts and Dashboards

It's time to chart our progress with an introductory series into the world of creating charts and dashboards in Excel. This month, we look at Scatter charts.

A Scatter chart, or XY chart as it is otherwise known, simply maps one set of numbers against another set, typically with the purpose of trying to establish whether there is any correlation amongst the data that may be difficult to see by reviewing the raw numbers alone. These charts are

used frequently by statisticians and scientists to visualise large volumes of data to see if any patterns exist.

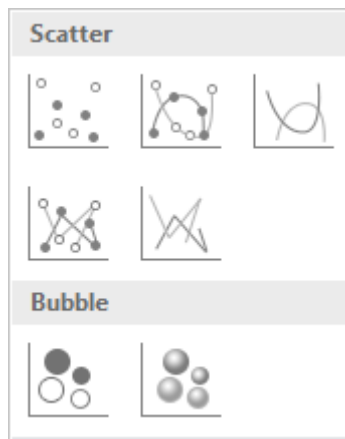
Assume that I have a set of data from TransProduct on customers' satisfaction and their transaction values as follows:

Response Number	#1	#2	#3	#4	#5	#6	#7	#8	#9	...	#575	#576
Transaction Value (\$)	1,027	560	142	999	1,069	340	816	753	179	...	474	923
Customer Satisfaction (0-5)	5.00	4.71	4.94	2.85	4.21	3.29	2.87	3.66	1.75	...	2.39	3.30

Looking at the table, it is already difficult – or almost impossible – for me to see any correlation between the two rows of data with 576 columns! (With regards to organising the data table, most people would not prepare the figures across a spreadsheet as illustrated above. Typically, this data should be arranged into two columns rather than two rows.)

With a Scatter chart, Excel is simply expecting two series so it can map one on the horizontal axis and the other on the vertical axis of the chart, so, it doesn't matter how the data is prepared – Excel can work it out anyway!

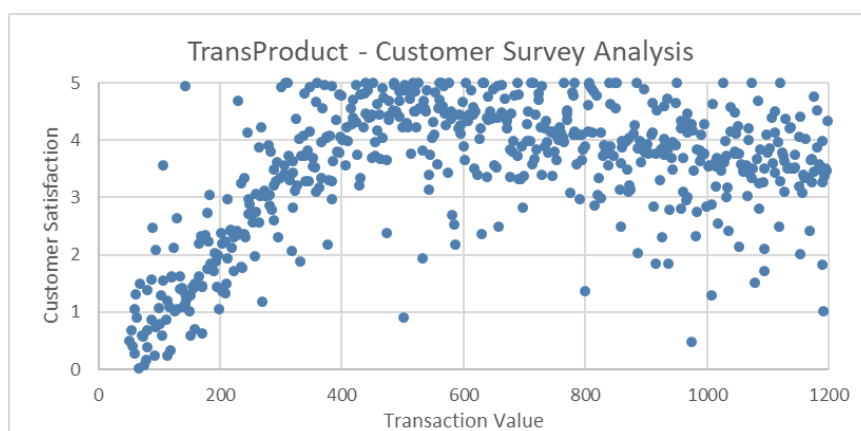
On highlighting the data, choosing 'Insert' then select the Scatter chart icon, so I may get a few recommendations:



There are seven variations of the Scatter graph. The first one is simply mapping each data point on the horizontal axis against the corresponding value on the vertical axis using a marker for each point. The next four are basically the same with the addition of smooth or straight lines, with or

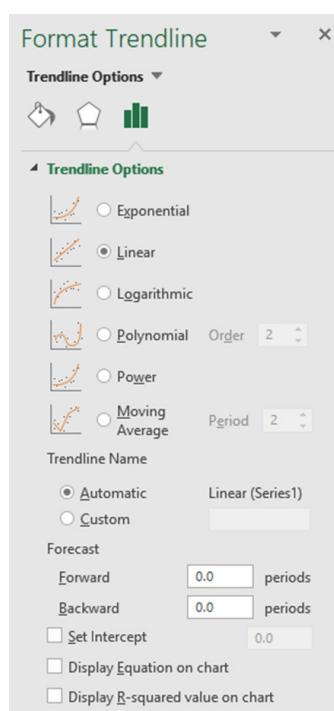
without markers. The last two under 'Bubble' section allows for a third data set, which is applied to determine the relative size of each bubble.

I choose the first variation and here is the corresponding chart:



It can be clearly seen from the chart that there is clustering occurring, starting very low in the bottom left corner, rising quite steeply as it moves along the horizontal axis, and then there is a spread of data points on the right side of the graph.

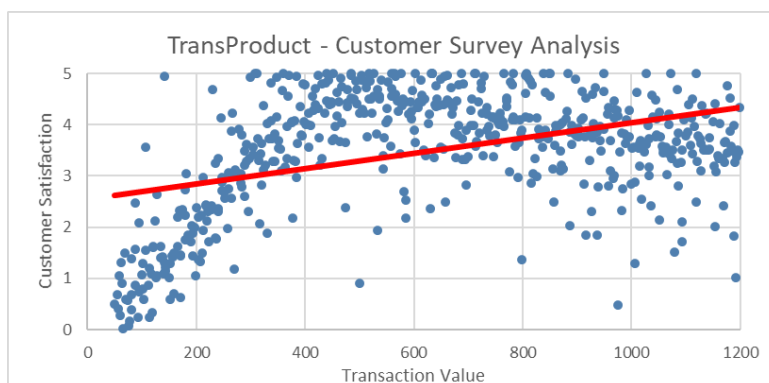
One of the strengths which I like in Excel is the ability to add trendlines to charts. A Scatter chart is one of the best charts to apply trendlines to for analysis purposes. To add a trendline, I right-click on the data series and choose 'Add Trendline'. This will add a linear trendline to the chart and the 'Format Trendline' panel will appear simultaneously:



'Trendline Options' offers me six different types of trendlines to pick, depending upon my purpose. Here I choose Linear, which is one of the most common trendlines used to create a single straight 'line of best fit' through the data. Another popular trendline is 'Moving Average' which

calculates the average of a specified number of previous figures and creates a line plotting these averages.

After I add a linear trendline to my chart and format it as solid red line, the chart now appears as follows:



I see the linear trendline does not align with my data: it only touches a few data points.

More next month...

Visual Basics

We thought we'd run an elementary series going through the rudiments of Visual Basic for Applications (VBA) as a springboard for newer users. This month, we continue looking at ListObjects to manipulate Tables within an Excel workbook in VBA featuring the Headers Range.

Let's not get ahead of ourselves! We begin with a Table:

Table Name:	Summarize with PivotTable	Insert Slicer	Export		
Table_PainRelief	Remove Duplicates				
Resize Table	Convert to Range				
Properties	Tools				
C6					
1	Column1	Column2	Column3	Column4	Column5
2	Nurofen	Ibuprofen	200mg		
3	Panadol	Paracetamol	500mg		
4	Disprin	Asprin	300mg		
5	Ponstan	Mefenamic Acid	250mg		
6	Solprin	Asprin	300mg		
7	Naprogesic	Naproxen Sodium	275mg		
8	Maxigesic	Paracetamol	500mg	Ibuprofen	150mg
9	Aspro	Asprin	300mg		
10	Mersynofen	Paracetamol	500mg	Ibuprofen	200mg
11	Panamax	Paracetamol	500mg		
12	Advil	Ibuprofen	200mg		
13	Voltaren	Diclofenac	12.5mg		
14	Herron Blue	Ibuprofen	200mg		
15	Herron Gold	Paracetamol	500mg		
16	Astrix	Asprin	100mg		

Aside from the incorrect spelling of "aspirin" (!), notice how our table has no header names? Excel has just given it the generic **Column + #** header name. How would one fix this in VBA? No need to bury your

head in the sand; it's no head-scratcher. Simply go head-to-head with the *HeaderRowRange* property of the *ListObject*. This can be used to retrieve the range of the header row.

```

(General)
Option Explicit

Sub Test ()

Dim MyTable As ListObject
Set MyTable = Range("Table_PainRelief").ListObject

Debug.Print MyTable.HeaderRowRange.Address

End Sub

```

Immediate
\$A\$1:\$E\$1

With a cool head, this then can be accessed directly to change the name of the headers. The Cells property can be used. Remember off the top of your head that *Cells* uses ([RowNumber], [ColumnNumber]) references –

the column ordinality of the Table will definite which header is changed. Let's run the following code and head toward a more meaningfully named Table:

```

(General)
Option Explicit

Sub Test ()

Dim MyTable As ListObject
Dim MyHeaderRow As Range

Set MyTable = Range("Table_PainRelief").ListObject
Set MyHeaderRow = MyTable.HeaderRowRange

MyHeaderRow.Cells(1, 1) = "Brand Name"
MyHeaderRow.Cells(1, 2) = "Primary Active Ingredient"
MyHeaderRow.Cells(1, 3) = "Primary Concentration"
MyHeaderRow.Cells(1, 4) = "Secondary Active Ingredient"
MyHeaderRow.Cells(1, 5) = "Secondary Concentration"

End Sub

```

Heads up! This is the result:

	A	B	C	D	E
1	Drug Nam	Primary Active I	Primary C	Secondary	Secondary
2	Nurofen	Ibuprofen	200mg		
3	Panadol	Paracetamol	500mg		
4	Disprin	Asprin	300mg		
5	Ponstan	Mefenamic Acid	250mg		
6	Solprin	Asprin	300mg		
7	Naprogesic	Naproxen Sodium	275mg		
8	Maxigesic	Paracetamol	500mg	Ibuprofen	150mg
9	Aspro	Asprin	300mg		
10	Mersynofen	Paracetamol	500mg	Ibuprofen	200mg
11	Panamax	Paracetamol	500mg		
12	Advil	Ibuprofen	200mg		
13	Voltaren	Diclofenac	12.5mg		
14	Herron Blue	Ibuprofen	200mg		
15	Herron Gold	Paracetamol	500mg		
16	Astrix	Asprin	100mg		

Success! But heads will roll if this table was sent out – ‘Drug Name’ is a misnomer, using ‘Brand’ name would be more accurate. If the column number is unknown, how does one rename the column? How do we make heads or tails about that?

Don't knock your head against a brick wall, use the *Find* method of Range to find the cell address of a particular name in the Header list and then the column position. Then reference the cell directly in order to change the header.

Option Explicit

```
Sub RenameTableColumns()  
  
    Dim MyTable As ListObject  
    Dim MyHeaderRow As Range  
  
    Set MyTable = Range("Table_PainRelief").ListObject  
    Set MyHeaderRow = MyTable.HeaderRowRange  
  
    Dim HeaderChangeAddress As String  
  
    HeaderChangeAddress = MyHeaderRow.Find("Drug Name").Address  
  
    Range(HeaderChangeAddress).Value = "Brand Name"  
  
End Sub
```

We are heading in the right direction. Hold your head up high with this correctly named Table:

	A	B	C	D	E
1	Brand Name	Primary Active Ingredient	Primary Concentration	Secondary Concentration	Secondary Concentration
2	Nurofen	Ibuprofen	200mg		
3	Panadol	Paracetamol	500mg		
4	Disprin	Asprin	300mg		
5	Ponstan	Mefenamic Acid	250mg		
6	Solprin	Asprin	300mg		

More next time if you want to stay a head of the game. *(Please stop – Ed.)*

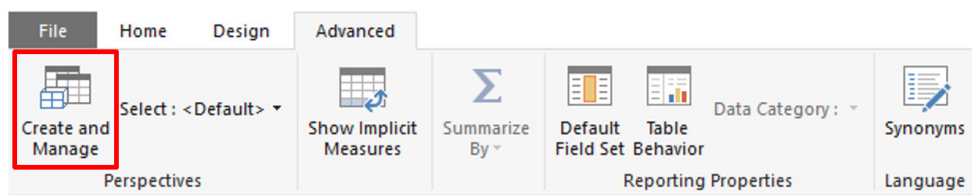
Power Pivot Principles

We continue our series on the Excel COM add-in, Power Pivot. This month, we discuss how to create and use Perspectives in Power Pivot.

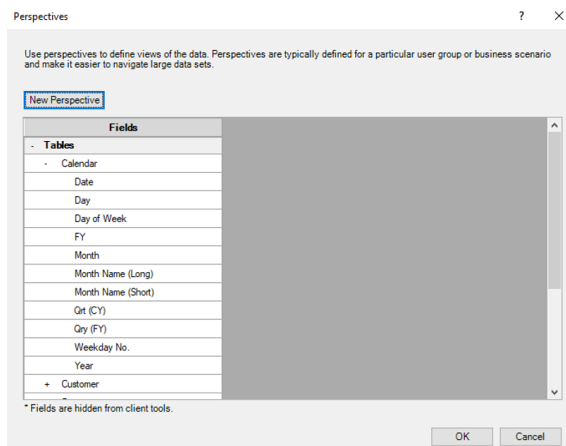
Perspectives is a custom view that we can define for a specific user group or business segment making it much easier to navigate through large data sets. We can include any combination of measures, columns and tables in a Perspective. Additionally, we can create multiple Perspectives for various reporting constituencies in a business. Different Perspectives

can even be used as a data source for PivotTables and reports.

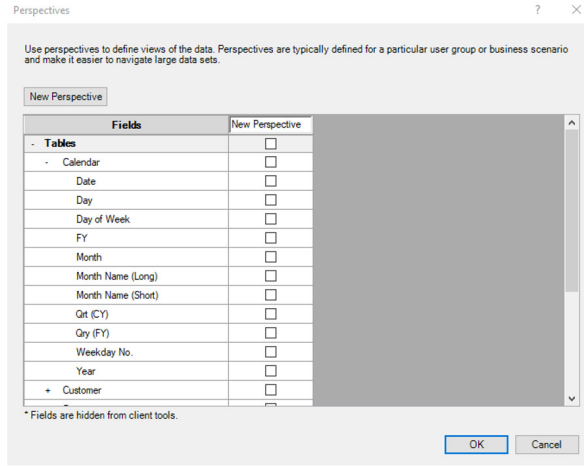
To create a Perspective, we first open the PowerPivot window navigate to the 'Advanced' tab on the Ribbon and click on the 'Create and Manage' option in the 'Perspectives' group:



This will bring up the Perspectives dialog:



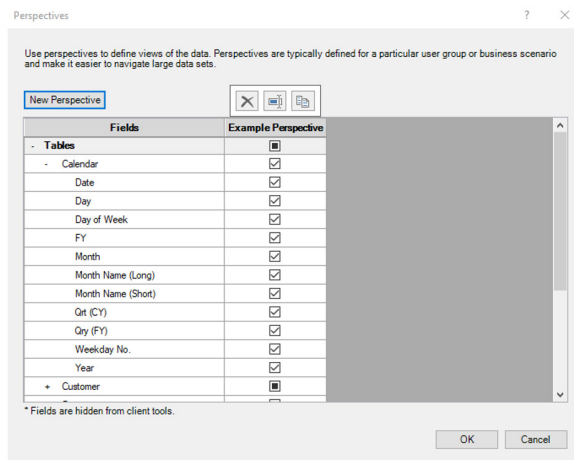
Click on the 'New Perspective' button:



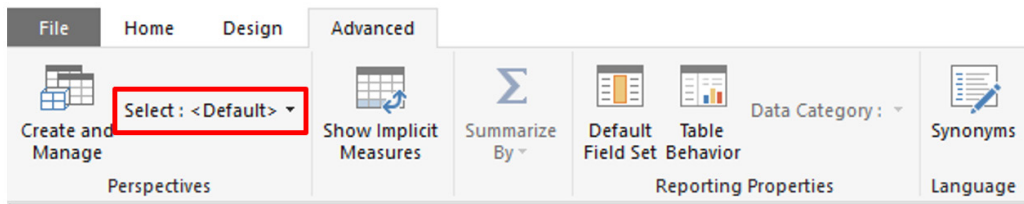
Here is where we can name the Perspective and select the tables, columns and measures that we wish to include in this Perspective.

a Perspective to bring up a small dialog box with three icons on it. The three options in the dialog box in order from left to right will be: delete, rename and copy the Perspective the mouse cursor is hovering over.

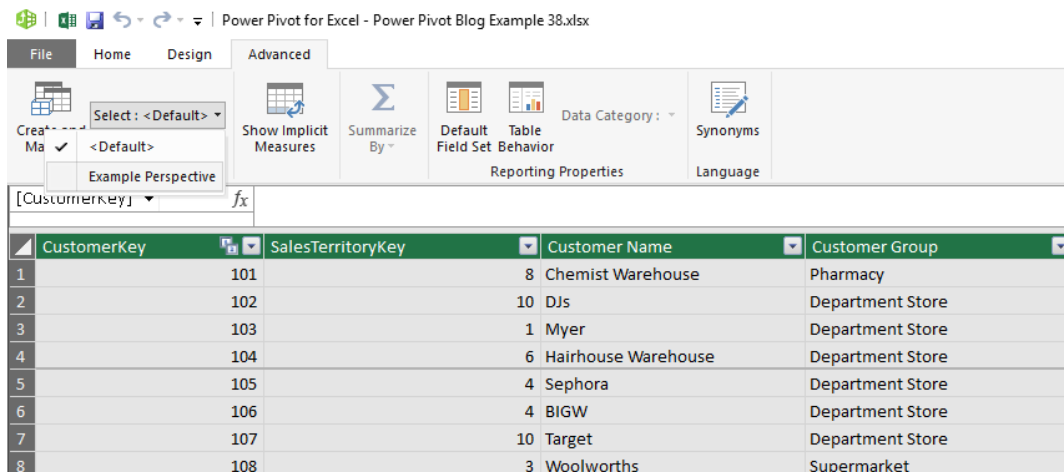
To delete, rename or copy a Perspective, hold the mouse cursor over



Once we have created our Perspective, click the 'OK' button. Our new Perspective can be found on the top left-hand corner of the PowerPivot window in the 'Perspectives' group.



Clicking on the 'Select: <Default>' button will bring down a drop-down list of all of our Perspectives allowing us to toggle between the Perspectives and the '<Default>' view (which contains all of the original data).



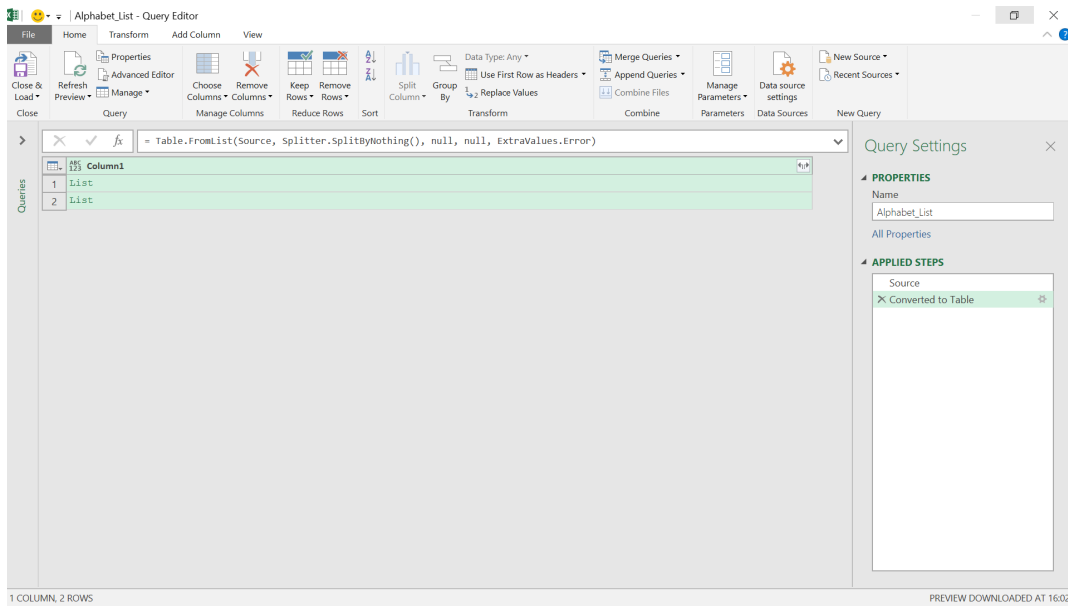
Remember to save in the PowerPivot window and the workbook to save the Perspective selection. Hopefully, this will provide give you with some tools to manage larger data sets and reduce the time spent scrolling through tables hunting for columns. It's a whole new Perspective!

More *Power Pivot Principles* next month.

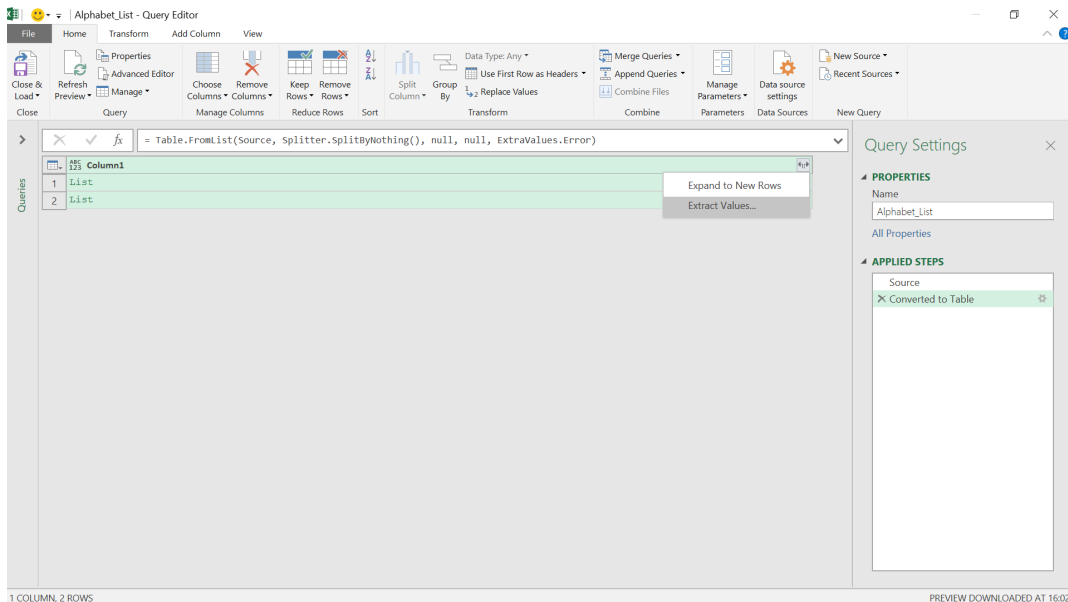
Power Query Pointers

Each month we'll reproduce one of our articles on Power Query (Excel 2010 and 2013) / Get & Transform (Office 365, Excel 2016 and 2019) from www.sumproduct.com/blog. If you wish to read more in the meantime, simply check out our Blog section each Wednesday. This month, we consider how extracting data from multiple lists can be confusing when nulls are involved.

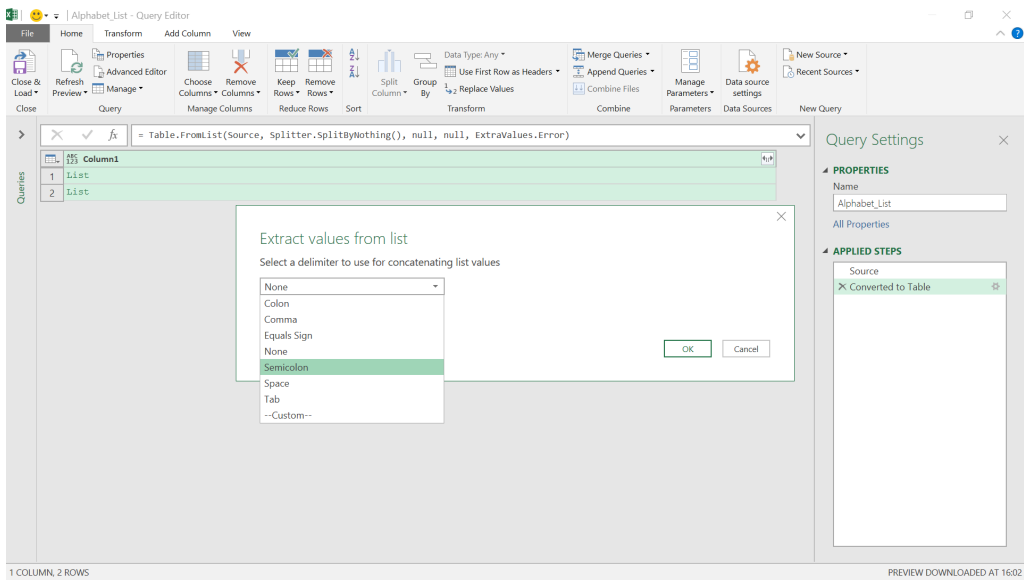
Consider a table of lists, which you need to extract. You want to keep the data aligned so that it makes sense. In real life, this would be more likely to be long references, but here, our chosen data will make the problem we're about to encounter more obvious.



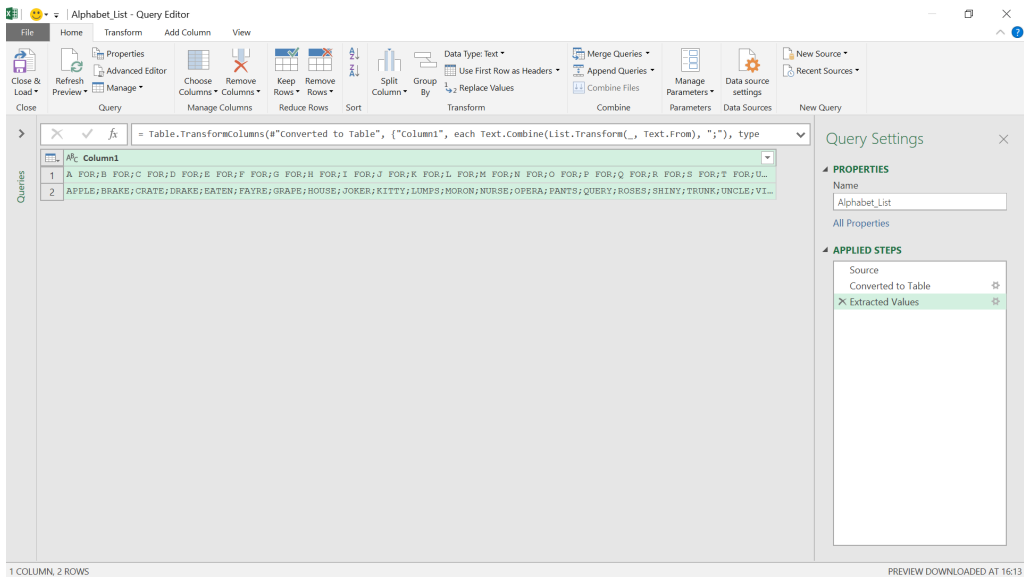
We want to extract the values from our list, so click on the little split line icon at the right of the **Column1** heading.



If you were to choose the 'Expand to New Rows' option, you would get a row for each value in each of my lists. We don't want this; we want to see our lists one above the other. Therefore, let's choose 'Extract Values':

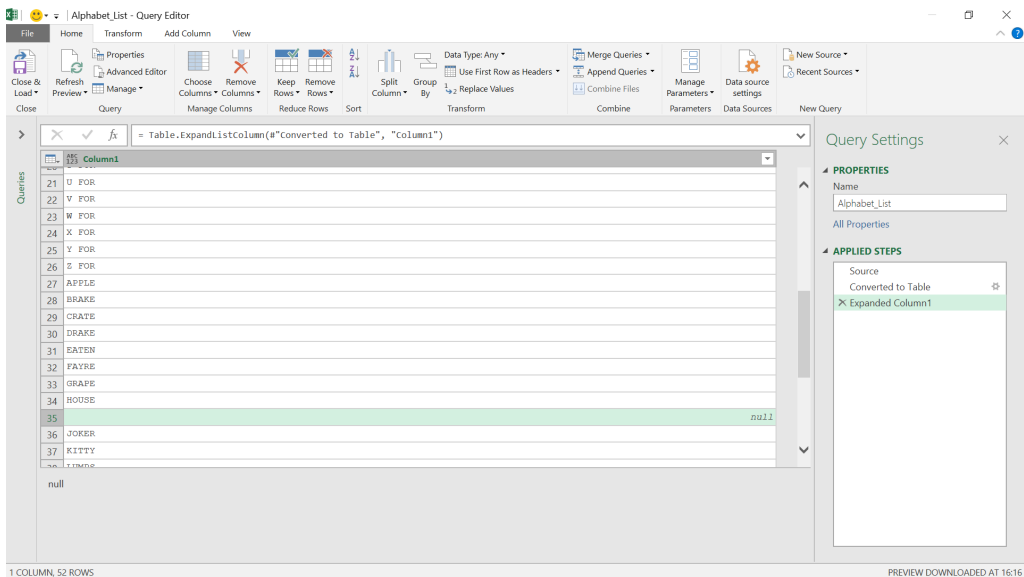


We may choose a delimiter to separate the values on my list, so let's choose to use a semicolon (;):



Our data starts off well. "A FOR APPLE" goes together properly, as does "E FOR EATEN". However, "P IS FOR QUERY" is not quite so good. If we check the data, we'll see that things start to go wrong at "I", which is definitely not for "JOKER"!

If we go back a step and look at our data in rows using the 'Expand to New Rows' option instead, we can check why the data doesn't make sense after "I":

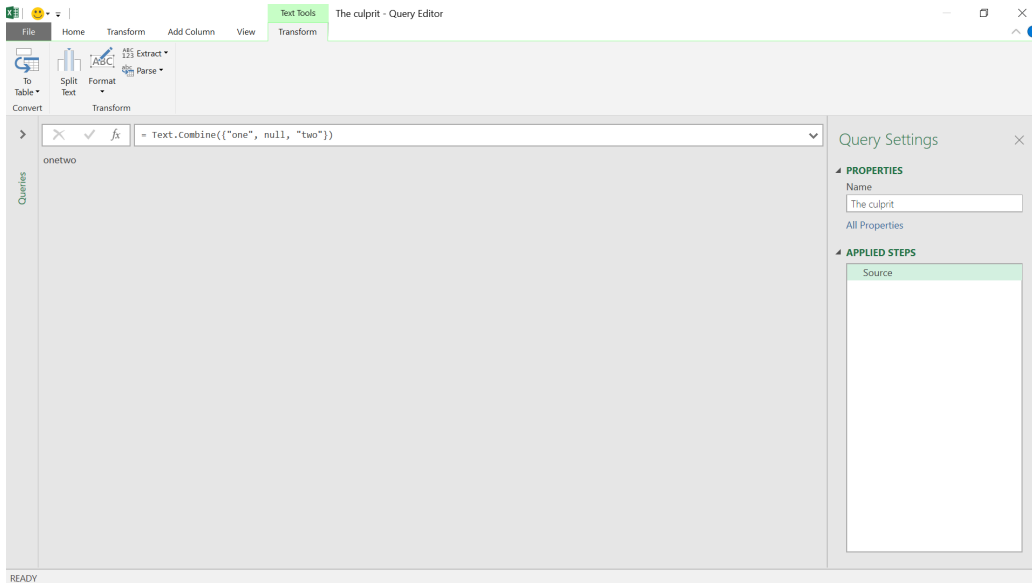


In the entry where the word beginning with 'l' should have gone, there is a *null*. The number of entries is right, and you can see the data is in the list, so why has Power Query ignored the *null* entry and misaligned our data?

To see the answer, we need to go back to my 'Extract Values' step and look at the **M** language behind it:

```
= Table.TransformColumns(#"Converted to Table",
{"Column1", each Text.Combine(List.Transform(_, Text.From), ";"), type text})
```

The culprit here is **Text.Combine**. We can test this with a simple scenario in another Blank Query:

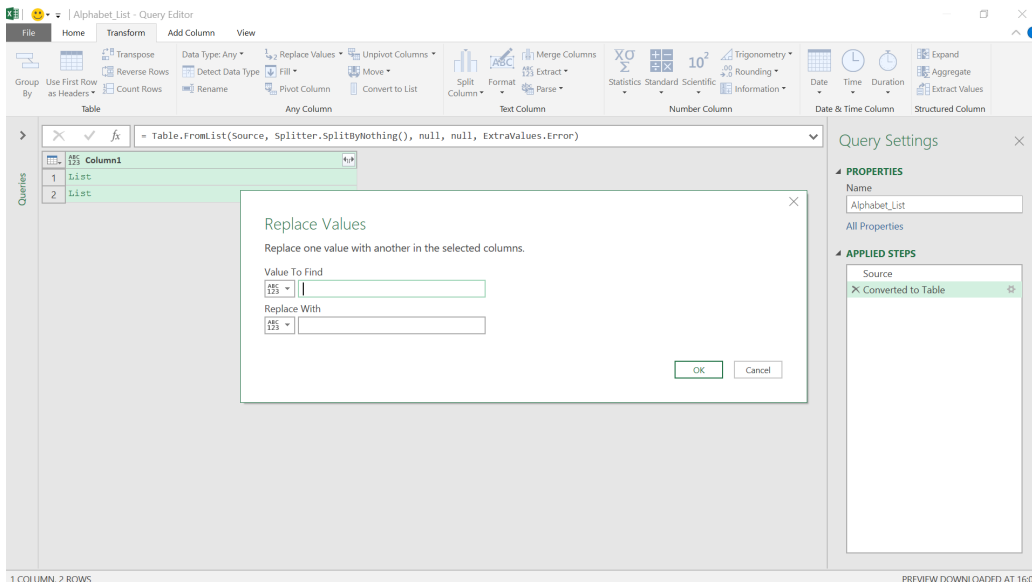


The **M** language used here is very simple, and it is easy to see what happens:

```
= Text.Combine({"one", null, "two"})
```

This combines the text in the list provided and ignores the *null* values. Therefore, the data will be misaligned. How we get around this would be to replace the *null* values in our lists with something else – and that something else really depends on the context. It could be 'N/A', spaces or whatever makes sense. In our alphabet example, we will replace null with "-----" until we can think of a good word for "l"!

It would be great if Power Query could use the Graphic User Interface (GUI) 'Replace Values' option for this:



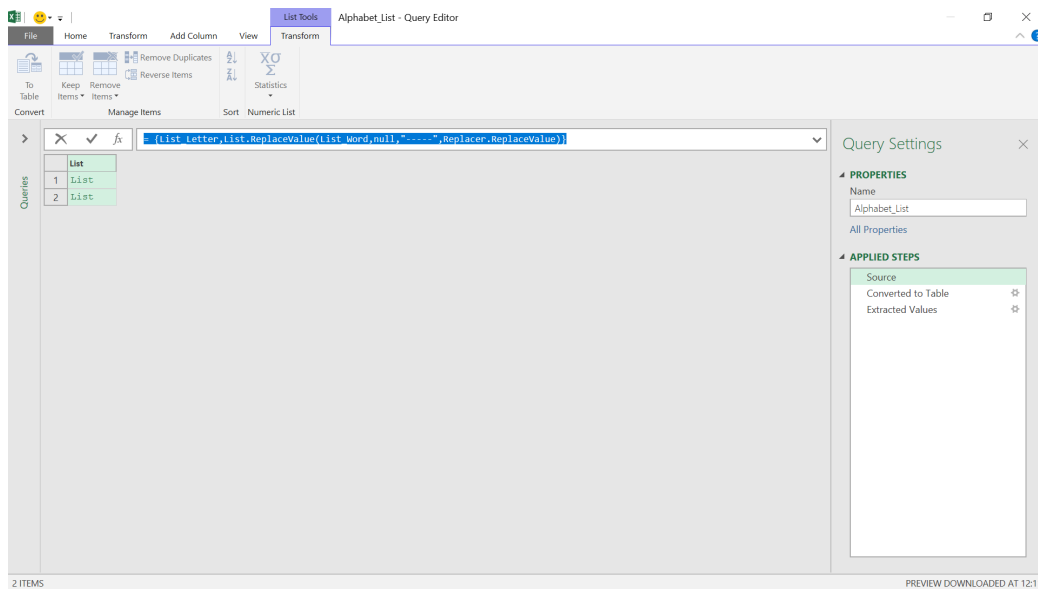
However, since the values are not currently visible, this will not work. We need some **List** functionality. Consider

```
List.ReplaceValue(list as list, oldValue as any, newValue as any,
replacer as function) as list
```

where **list** is the list to modify, **oldValue** is the value to replace (the *null* in this instance), and **newValue** is the value to replace it with ("-----"). The

argument **replacer** is used to tell Power Query what kind of values are being replaced, and in this case, we would use **Replacer.ReplaceValue**.

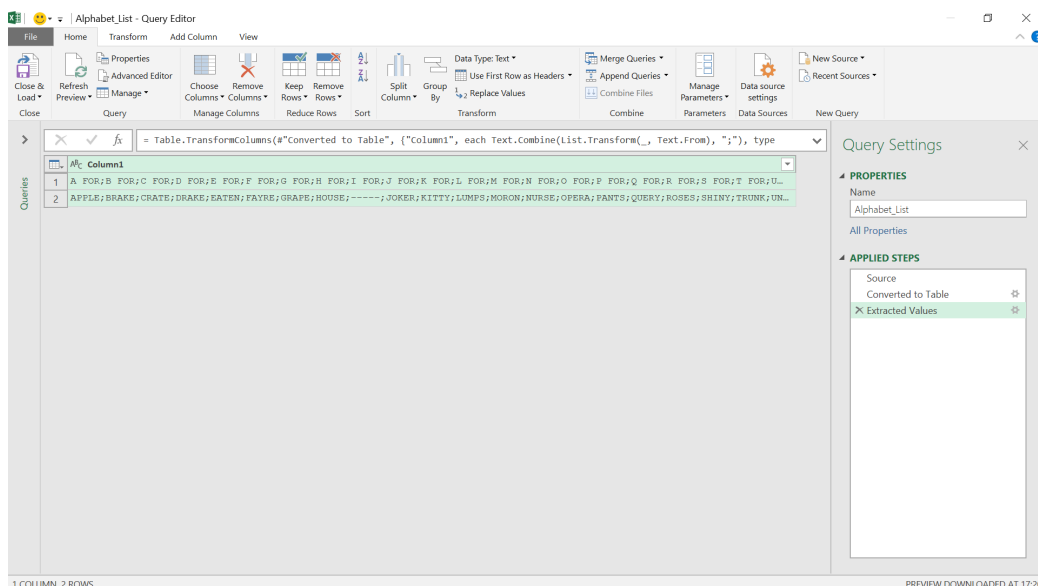
We need to add this step before we start manipulating the lists. In practice, it would be best to apply this to any list where there is a risk of *null* values being present, but we'll add this to our second list, as that is the one causing us problems.



Since the lists are defined in our source, the M language we need for the second list is:

= {List_Letter,List.ReplaceValue(List_Word,null,"-----",Replacer.ReplaceValue)}

Now when we execute the query again, we'll see the lists aligned correctly, viz.



More next month!

Power BI Updates

This month's updates see a new sparklines Preview feature, several new Format Pane updates, e-mail subscription access via Admin API and numerous other useful features. In detail:

Reporting

- Sparklines Preview
- New Format pane updates in Preview

Data Connectivity and Preparation

- Azure Data Explorer (connector update)
- FHIR (connector update)
- Dremio Software (connector update)
- BQE Core (connector update)
- TIBCO Data Virtualization Generally Available
- Exasol (connector update)
- Starburst Enterprise (connector update)

Service

- Administration and governance:
 - o Email subscriptions access via Admin API
- Allow sharing links to include your changes to the report
- Announcing Preview of Hybrid Tables in Power BI Premium
- Deployment pipelines: assign a workspace to all pipeline stages

Mobile

- New Report default view
- Recommended Content supports hints (Android and iOS)
- Mobile support for Goals: latest updates

Embedded Analytics

- Support Power BI dataset as a data source for embedding paginated reports
- Client API updates: Themes
- Azure Power BI Embedded A7 and A8 capacity nodes self-service

Sparklines Preview

Sparklines come into Preview as part of this month’s update. Just like in Excel, sparklines are tiny charts shown within cells of a table or matrix that make it easy to compare a large number of trends quickly. You

Developers

- Improve your matrix performance with the new 4.1 API

Visualisations

- New visuals in AppSource
- Shielded HTML Viewer by Nova Silva
- graphomate charts 2021.4
- Zebra BI Cards 1.0 by Zebra BI
- Leverage process mining through Business Intelligence by PmBI
- Drill Down Combo Bar PRO by ZoomCharts
- Paginated report visual Generally Available
- Charticator (version 1.2.0)

Other

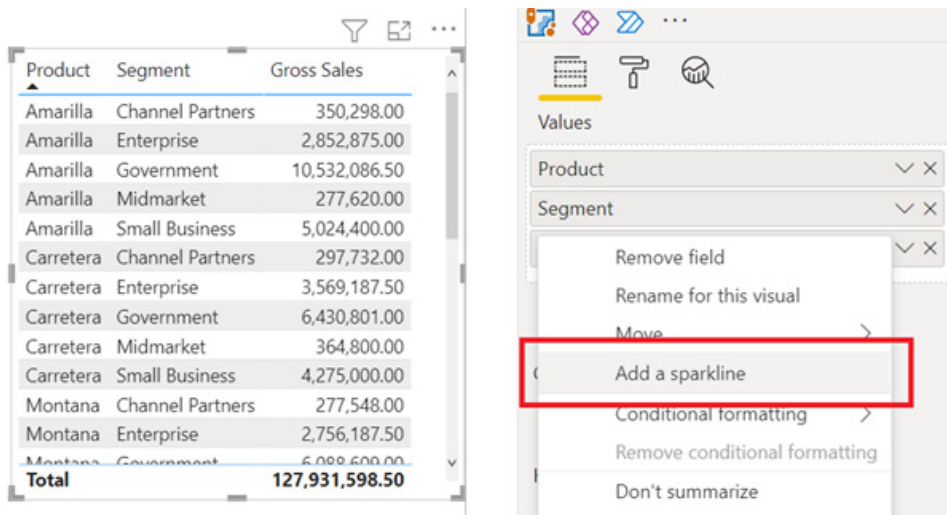
- Visualise your data quickly from Power Apps and Dynamics 365 apps.

Let’s go through each in turn.

can use sparklines to show trends in a series of values, such as seasonal increases or decreases, economic cycles or to highlight maximum and / or minimum values.

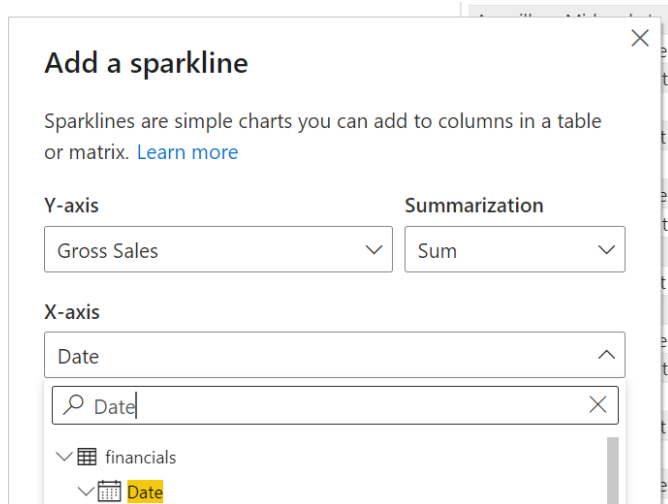
Segment	Channel Partners	Enterprise	Government	Midmarket	Small Business	Total
Product	Sales	Sales by Date	Sales	Sales by Date	Sales	Sales by Date
Amarilla	317,643.00		2,643,607.50		9,942,899.11	
Carretera	282,838.68		3,203,708.13		6,080,944.08	
Montana	261,844.56		2,614,843.75		5,548,936.02	
Paseo	454,514.40		5,267,860.00		14,882,230.70	
Velo	182,924.04		3,581,237.50		7,813,422.05	
VTT	300,828.96		2,300,437.50		8,235,828.71	
Total	1,800,593.64		19,611,694.38		52,504,260.67	

With this Preview feature enabled, you may add sparklines to both your table and matrix visuals through the field well. Simply open up the context menu for one of your numeric fields and select ‘Add a sparkline’.

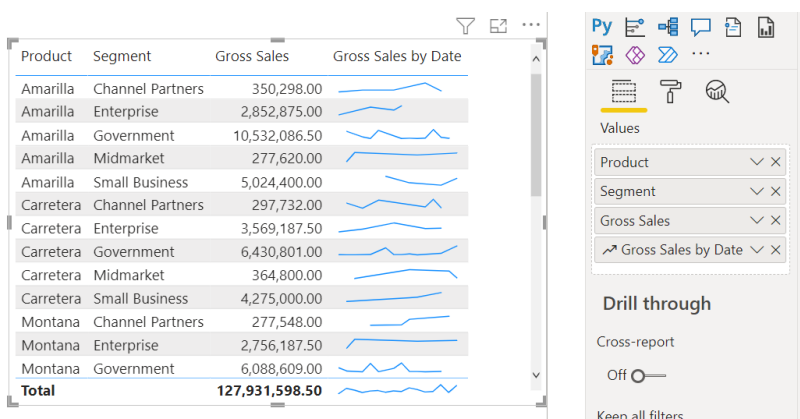


A dialog will appear to let you configure the details of your sparkline. The numeric field you started with will be pre-populated for the y-axis, but you are able to change both the field and ‘Summarization’ (sic) type

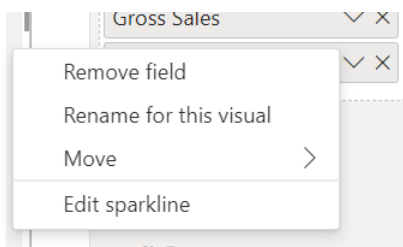
if needed. You’ll also need to select a field, such as a date column, to use as the x-axis of the sparkline.



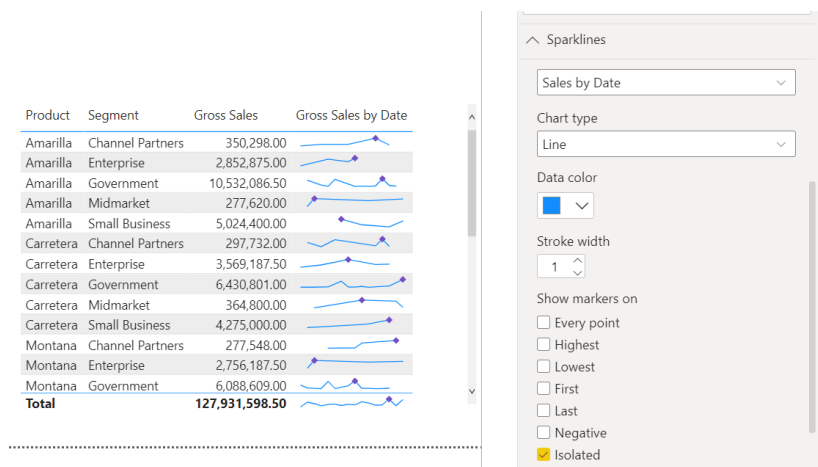
Once you click 'Create', your sparkline will be added to your table or matrix as a new column automatically.



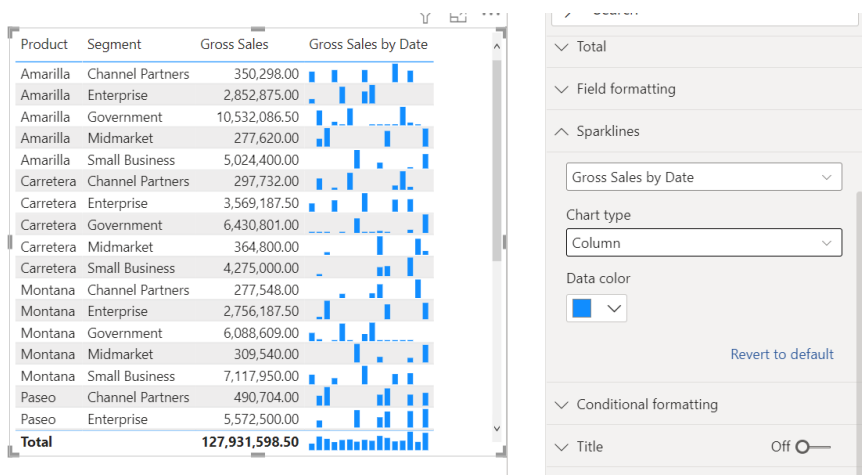
You can always edit the sparkline's properties through the 'Edit sparkline' option in the value's context menu.



You can use the Sparklines card in the Formatting pane to modify the sparkline's line and marker formatting. You're also able to change the line colour and width, add markers for different value types (highest, first, last, etc.), and change the marker size, colour and shape.



This is also where you may change the chart type from a line to a column chart.



Power BI supports up to five sparklines per visual and will display up to 52 points per sparkline. For performance reasons, the maximum number of columns in a matrix will be limited to 20 when sparklines are

on. Note that although sparklines will be supported on Azure AS, it is not supported in any current on-prem SSAS but should work with future SSAS releases.

New Format pane updates in Preview

Since the new Format pane came out in Preview, there have been a few gripes. To address concerns raised regarding the extra clicks needed to find settings, the context menu 'expand all categories' now expands all cards and sub-categories to meet what everyone wants.

The following missing settings have also been re-added:

- Action card for buttons, shapes and images
- responsive toggle and data limit settings for **General** -> **properties**
- global text size for table / matrix
- background for **Scatter chart** -> **category labels**
- **Analytics** -> **Ratio line** for scatter charts.

Numeric input truncation and incorrect red borders bugs have both been fixed too.

Azure Data Explorer (connector update)

The Azure Data Explorer connector has been updated with the following changes:

- fixed support for authentication against Log Analytics / Application Insights / Azure Monitoring resources
- support folding of Bitwise Shift Left / Right operations
- fixes to Text.PositionOf folding
- accepting batch requests (only the last result set will be used).

FHIR (connector update)

The FHIR connector has been updated with a fix for an issue where data was not filtered when additional query parameters were appended.

Dremio Software (connector update)

The Dremio Software connector has added the ability to authenticate using Azure Active Directory (AAD) in addition to the existing username and password, or username and personal access token methods.

BQE Core (connector update)

The BQE Core connector has been updated. There is now more data access via the BQE CORE connector, including:

- contacts
- CORE CRM
- Fund Transfers.

BQE CORE now has a Power BI connector directly through the Microsoft Power BI app. This connector gives users the ability to import their raw CORE data into the Power BI application, as well as transform it into meaningful, centralised, visual and interactive insights for their

business. While the connector was offered previously through training programmes, this installation from the Power BI app will help CORE customers create custom dashboards and reports, and ultimately make more informed business decisions.

TIBCO Data Virtualization Generally Available

This connector is now Generally Available.

Exasol (connector update)

This update enables the 'View Native Query' option.

Starburst Enterprise (connector update)

The Starburst Enterprise connector has been updated. There is now:

- support for version 2.0.0+ of the Starburst ODBC driver
- support has been removed for the Presto ODBC driver (prior to ODBC version 2.0.0).

Administration and governance

EMAIL SUBSCRIPTIONS ACCESS VIA ADMIN API

Email subscription to reports and dashboards is a feature that makes it easy for users to stay current regarding their key data. To make this feature more useful, better administration visibility has been applied to these subscriptions so that users may stay on top of governance and control. In the past, it was nigh on impossible to support scenarios such as magnitude assessment of PII data breaches and leaks due to

subscriptions, particularly in the Healthcare sector. It required going through the user interface and was cumbersome and inefficient. You could also gain insight into subscriptions by analysing the audit logs, but audit logs only go back 90 days at most. Therefore, this release for Public Preview provides a set of Administrator APIs that return lists of all the email subscriptions that have been created in the entire tenant.

Allow sharing links to include your changes to the report

The new link sharing experience provided in this update now supports the option to share your explored version of the report through 'Include my changes', viz.

Send link
IT Spend Analysis Sample

People in your organization with the link can view, share, and build new content

Enter a name or email address

Add a message (optional)

Include my changes ⓘ

Send

Copy link Mail Teams

You may use this setting to create a shared view of the report that includes any of the filters or changes you've made to the report within read mode.

The 'Include my changes' setting includes any changes to the following:

- filters (Filter pane or URL filters)
- Slicers
- Personalize visuals
- cross-filtering or cross-highlighting
- drill down or drill up
- applied bookmark
- Spotlight.

Announcing Preview of Hybrid Tables in Power BI Premium

This release now supports Hybrid Tables in Power BI Premium, including an enhancement to incremental refresh: incremental refresh augmented with real-time data. Now users may see largescale improvements in performance in import mode and the latest data changes in the data warehouse reflected in user reports without having to perform a dataset refresh.

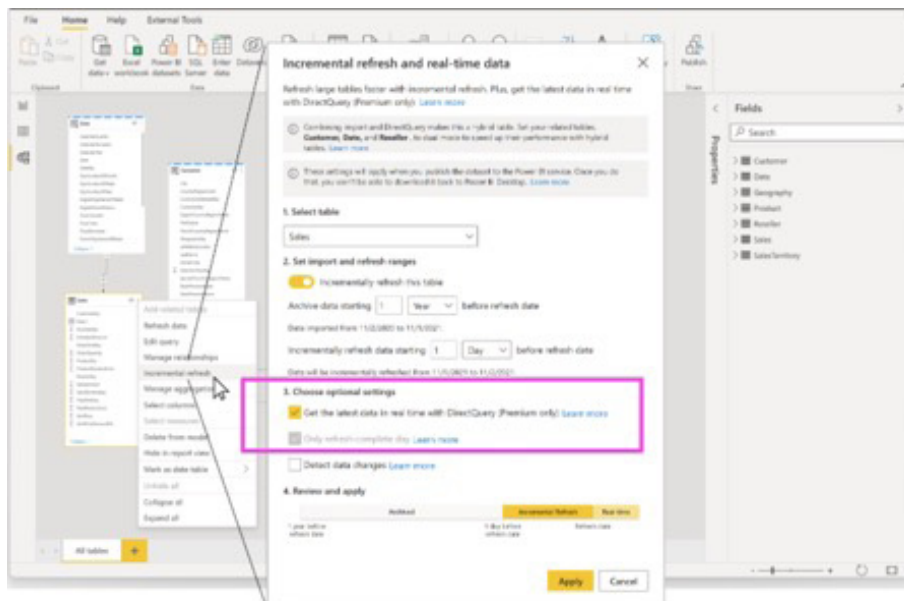
Up until now, dataset creators sometimes had to make hard trade-offs between query performance and the currency of data. Import mode provides better performance, yet data freshness poses challenges if new data arrives at a very high cadence. It is both resource intensive and inefficient to import data into a dataset in very short intervals. DirectQuery mode, on the other hand, delivers data freshness, yet at the expense of report performance. Because Power BI doesn't import the data but translates the report queries into data source queries, the latest

data changes are quickly picked up, but the query / response roundtrips between Power BI and the data source take time and slow down the reports.

As the following diagram illustrates, a Hybrid Table may help to strike the right balance between query performance and data freshness. A Hybrid Table is essentially a (large) table that has one or multiple import-mode partitions as well as another partition in DirectQuery mode. If the DirectQuery partition is sufficiently small in comparison to the import-mode partitions, the query / response roundtrips between Power BI and the data source should still be reasonably fast while access to the bulk of the data is already super-fast in import mode. Import and DirectQuery data is presented to users as a single table with business definitions and calculations.

Granularity	Name	Row Count	
Year	2011	295,489,717	} Archived: Import
Year	2012	297,678,498	
Year	2013	295,575,442	
Year	2014	292,477,875	
Year	2015	297,780,469	
Year	2016	294,060,081	
Year	2017	300,419,682	
Year	2018	296,541,108	
Year	2019	292,787,420	
Year	2020	299,273,979	
Quarter	2021Q1	74,135,277	} Incremental refresh: Import
Month	2021Q104	24,939,498	
Day	2021Q10501	820,805	
Day	2021Q10502	826,885	} Real time: DirectQuery
Day	2021Q10503	821,043	
Day-DirectQuery	2021Q10504-DQ	271,110	
Total		3,063,898,887	

The following screenshot shows how to configure an incremental refresh policy with real time in Power BI Desktop. Having published the dataset to a Premium Per User (PPU) workspace or a workspace on a Premium capacity, Power BI will automatically apply the refresh policy and partition the table as a Hybrid Table during data refresh.



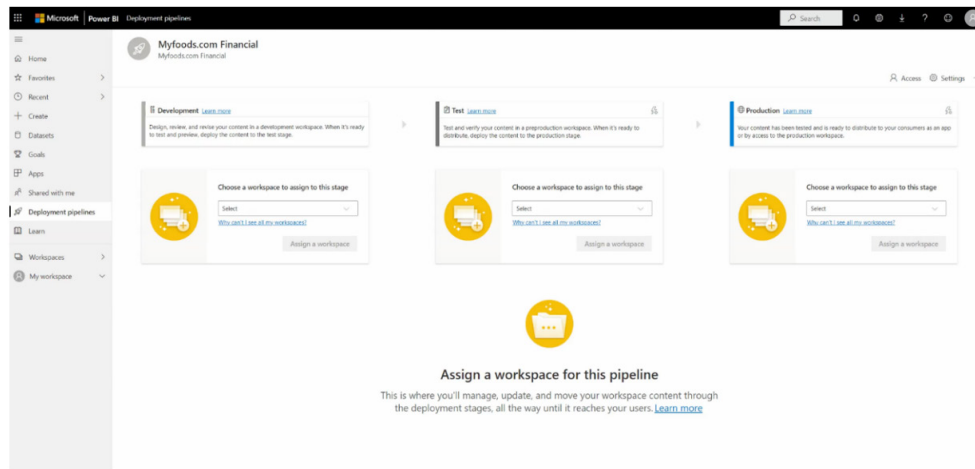
Deployment pipelines: assign a workspace to all pipeline stages

Starting in early December, deployment pipeline users may assign workspaces to all the pipeline stages. This allows maximum flexibility for Power BI creators that want to use the content they already built and manage it through deployment pipelines.

Some users already built their own release process using development / testing / production workspaces. They can now 'lift-and-shift' that

process and place it in a pipeline, to make a more productive and reusable release process.

It can also be helpful for users who don't have permission to create new workspaces. Now, they may ask IT to prepare the relevant workspaces for them and just assign these workspaces to the pipeline and start working in them immediately.



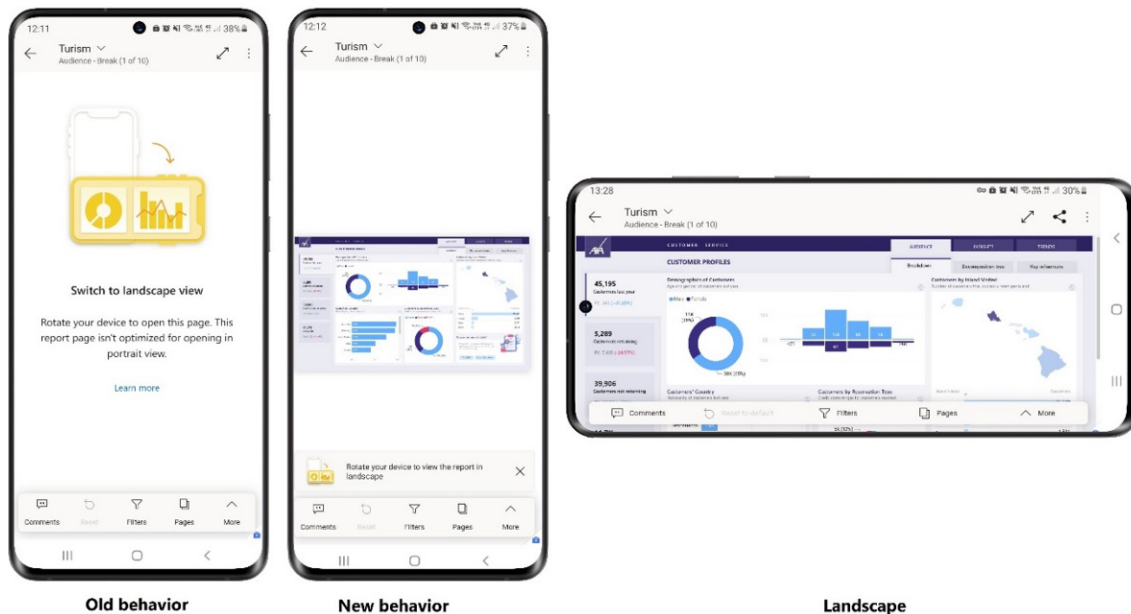
To make the experience smooth and easy, after assigning two workspaces, the pipeline will automatically match items between the workspaces and compare them. The matching is done based upon the item type and its name. Whenever a match is found, it means that deploying the artifact from the source stage to the target stage will override and update its match.

Note that the matching operation is done only upon assigning a workspace. If for some reason you want to 'match' specific items between stages in an already assigned workspaces, the only way to do that is by unassigning one of the workspaces and assigning it back, to trigger the matching between items.

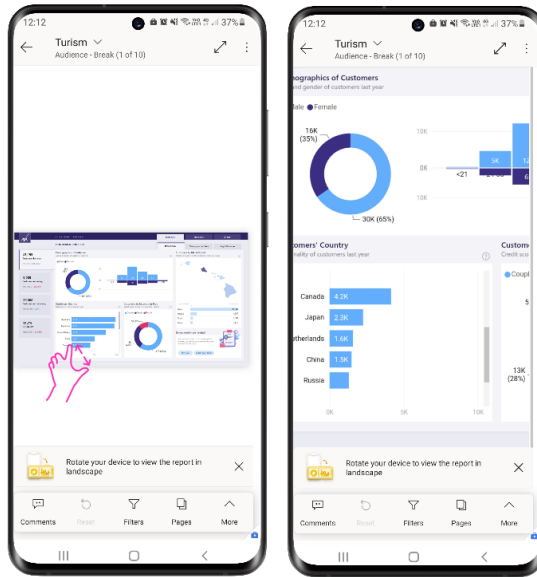
New Report default view

Most of us hold our phones in portrait orientation when we look at them, but for the most part, Power BI reports are designed for viewing in landscape orientation on computer screens. As a result, when you look at a report on your phone, the mobile app often requires you to tilt your phone on its side in order to see the report (unless the report has a mobile-optimized layout). This can be very annoying, especially when some of the report's pages are optimized for viewing on mobile devices and others are not.

In this Power BI Mobile update, Microsoft has removed this constraint. Now, you may choose to view any Report page (mobile-optimized or not) in the orientation you want. If the report page you're looking at does not have a mobile-optimized layout but you want to hold your phone in portrait orientation, you'll just get a smaller landscape view of the report.



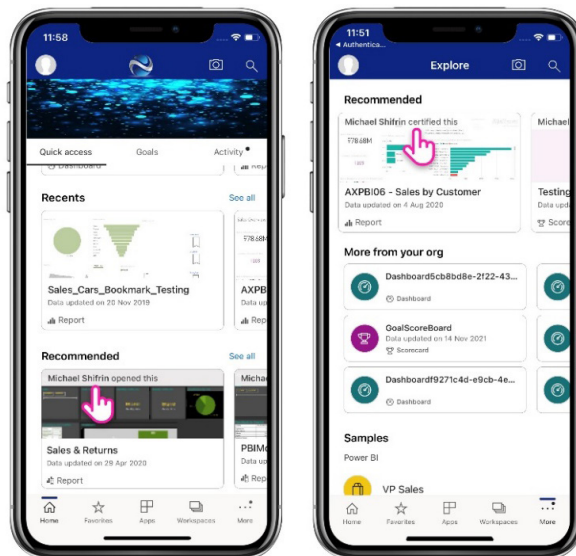
You may always tilt your phone to landscape orientation for a closer look at your data, but if all you need is a quick look at the numbers, you can quite easily stay in portrait orientation and just use pinch and zoom.



Recommended Content supports hints (Android and iOS)

Microsoft has added hints to the items recommended for you by Power BI to help you understand why the items were picked for you as recommended content. Being able to see who opened, certified

or published an item can help you decide whether the item might be relevant for your work.



Mobile support for Goals: latest updates

The mobile apps for iOS and Android now support Goal-level permissions and scorecard column customisations that have been set up by the scorecard author.

Goal-level permissions in Power BI enable scorecard authors to grant different levels of access to different users for specific goals. For instance, some users may be permitted to add notes to a goal and / or to update the goal's status and current value, while other users may only

be allowed to view specific goals within a scorecard. The mobile apps now support Power BI's goal-level permissions model.

Scorecard creators may also determine which columns consumers can see, ensuring that each scorecard shows only those columns that are appropriate to their scenario. The Power BI Mobile apps support such customisations, and will adjust the fields shown, and the check-in data that is required, accordingly.

Support Power BI dataset as a data source for embedding paginated reports

With Power BI embedded analytics, independent software vendors (ISVs) and customers may create Power BI content that displays paginated reports in a fully integrated and interactive application. They can embed paginated reports using the solution that works best for you, embed for your customers or embed for your organisation. Until now, Power BI dataset was not supported as a data source for those scenarios, but now, this has been enabled.

When embedding a paginated report with a Power BI dataset as a data source for the embed for an end users scenario, you need a service principal and a multi-resource embed token, that contains a new parameter called "XMLA permissions". When the value of this parameter is "Read Only", ISVs and customers can control the access to XMLA endpoints.

Power BI datasets with data sources that require single sign-on (SSO) are not supported yet, but that's coming soon.

Client API updates: Themes

Power BI report themes allow you to apply design changes to an entire report. For example, you can include your organisation's logo or change icon sets to match your corporate identity. Developers using Power BI embedded analytics also can embed a report with a custom theme applied instead of the default theme for the report. Applying a theme this way loads the same report with different themes to different users according to their settings. Microsoft has added the `getTheme` operation to allow ISVs obtain the embedded Power BI Report theme and apply it to their own web app.

Azure Power BI Embedded A7 and A8 capacity nodes self-service

The Embedded A7 capacity is similar to a P4 capacity node, carrying 64 v-cores and 200GB RAM. The Embedded A8 capacity is similar to a P5 capacity node, carrying 128 v-cores and 400GB RAM. Both capacity

In addition, two properties were added to the Page class:

1. Background: the background colour of the page and its transparency
2. Wallpaper: the wallpaper colour of the page and its transparency.

Both can be returned from the methods `report.getActivePage()` or `report.getPages()`.

Improve your matrix performance with the new 4.1 API

With the new 4.1 API, users may can fetch the custom matrix data on expand / collapse in a similar way to the built-in matrix. Data will be fetched on demand, to optimise the performance of the custom matrix.

nodes could be purchased by special request only. Microsoft has enabled their purchase in self-service, either through the Azure Portal or using an ARM API.

In order to use this feature, you need to specify the `selectionID` of the selected node when calling the `expand / collapse` API, which will be triggered by the `selectionManager.toggleExpandCollapse()` method.

New visuals in AppSource

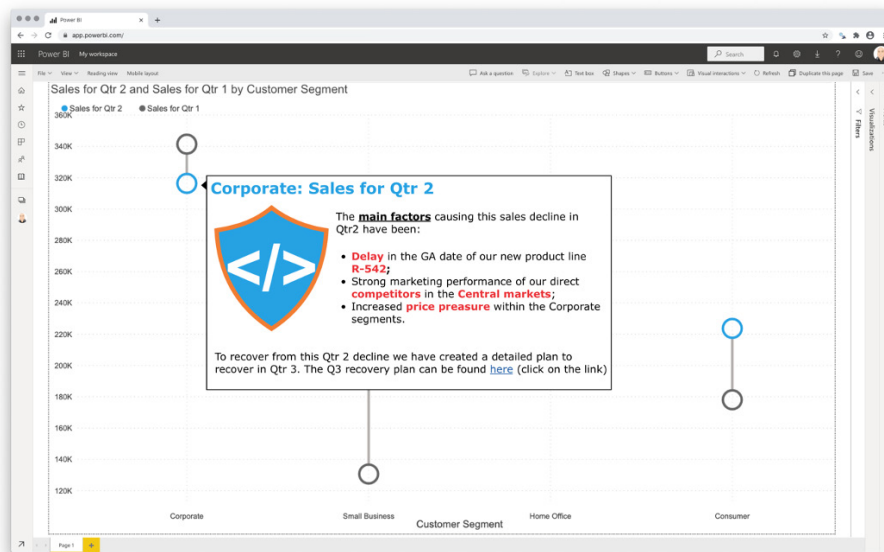
The following are new visuals this month:

- Data map by dTyp
- ProcessChampion by Argent Logic
- Ultimate UpSet Plot by Samuel Gratzl
- LeapLytics – Traffic Light by LeapLytics GmbH
- Deneb: Declarative Visualization in Power BI by Daniel Marsh-Patrick
- Inforiver Standard Matrix / Table by Lumel
- Icon Pack by TME AG
- HTML & CSS Viewer by K Team Solutions GmbH.

Shielded HTML Viewer by Nova Silva

All HTML rendered through the Shielded HTML Viewer is sanitised to keep your data safe. Sometimes, this sanitation process may be too strict. Based on user feedback, it's been noted that the use of images

has become a victim of this strict sanitation process. Therefore, support for images to the Shielded HTML Viewer has been added, without the need for risky external links.



As of version 2.0, the Shielded HTML Viewer (in the image above used in the tooltip of a dumbbell chart) supports inline SVG and Data URL images. This way, you can include any inline images in your Power BI reports and dashboards. Version 2.0 also includes support for the NAV-tag: now you can include HTML navigation menus to your reports.

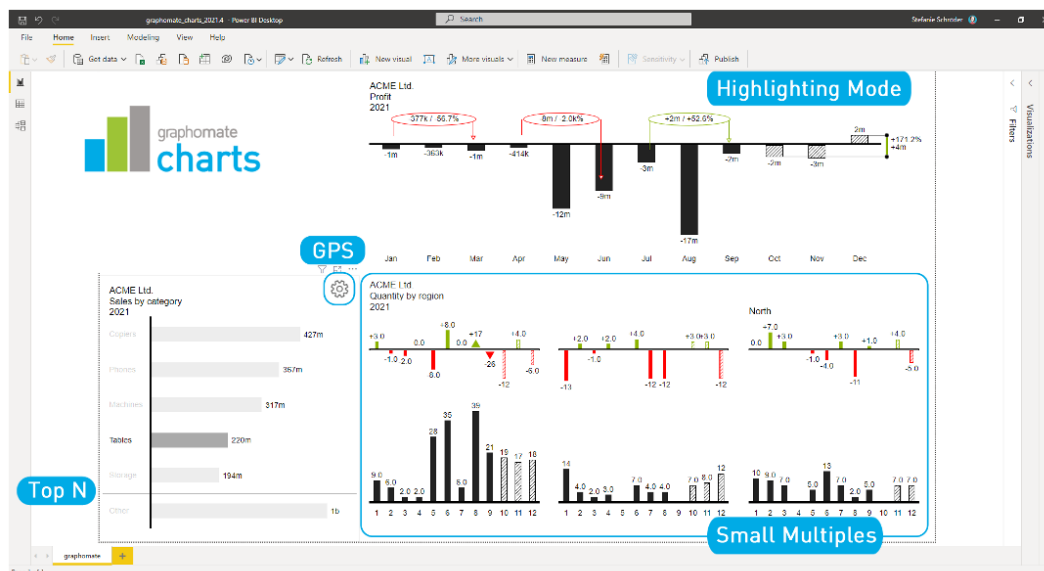
Furthermore, all functionality is available through the standard Power BI interface, so there is no need to learn a new interface. This may be downloaded from the AppSource.

graphomate charts 2021.4

The graphomate charts are designed to support users in creating meaningful reports and dashboards. They are compliant to the International Business Communication Standards (IBCS).

Here is a small selection of the latest features / updates:

- freely definable variance graphs, without a limit on the number of graphs
- detailed number formatting, which automatically adapts the language of your operating system
- show small multiples of a chart through drill down to another dimension
- create your own data scenarios, e.g. a certain format for actual or planned data and assign them to each datapoint.



With the 2021.4 release, the performance of the visual and new features has been improved. These updates include:

- quick access to the graphomate property sheet (GPS) via the gear wheel. You may now use the integrated search to quickly find and change properties
- you can highlight your findings with single Highlight and Multi-Highlights
- display measures without dimensions in the categories
- better sorting and setting options on the Format Pane of Power BI for the graphomate extensions.

Again, this may be downloaded from the AppSource.

Zebra BI Cards 1.0 by Zebra BI

There is a brand-new Zebra BI visual: Cards. With this new visual, you may now present your KPIs or your scorecard in a completely new way. Advanced features like focus mode, drill-through, dynamic comments and grouping allow better narration. Key functionalities include:

- See what's happening immediately:
 - show up to three comparisons
 - time series for trends
 - scaling of variance icons
- Dive deeper into data:
 - dynamic comments
 - drill through
 - focus mode
 - scaling option for charts on separate cards
 - invert individual cards for negative KPIs
- Adjust the design:
 - resize each Card based on the KPI's importance
 - drag and drop to reorder Cards
 - chart slider on each Card
 - number formatting option for each Card
 - axis break for all charts
 - Custom colours
 - Card and row layout.



With the Zebra BI Cards visual, you may combine its functionalities with charts and tables visuals.

Leverage process mining through Business Intelligence by PmBI

Process mining is the field that supports business process data analysis or the analysis of any dataset with a sequential concept for discovering the overflows, bottlenecks, delays, different unique behavioural patterns and fraudulent activities, etc.

Business Intelligence by PmBI provides a simple tool for creating Power BI reports and dashboards to reveal many hidden valuable insights obtained from your real business processes and integrated data.

Drill Down Combo Bar PRO by ZoomCharts

Drill Down Combo Bar PRO allows you to combine multiple chart types and apply customisation options for all aspects of the report. Interactive drill downs and on-chart interactions make data exploration simple.

Drill Down Combo Bar PRO is mobile friendly and supports interactions, selections, custom and native tooltips, filtering, fetching more data, bookmarks and a context menu.

Main features include:

- cross-chart filtering: select data points on multiple charts instead of using slicers
- on-chart interactions: click, pan, zoom in / out to explore multiple drill-down levels
- stacking and clustering: choose from various stacking options
- static and dynamic thresholds: set thresholds to demonstrate targets or benchmarks
- full customisation: customise x and y axes, legends, stacks, ToolTips, fill settings and / or outlines.

Some of the more popular use cases include:

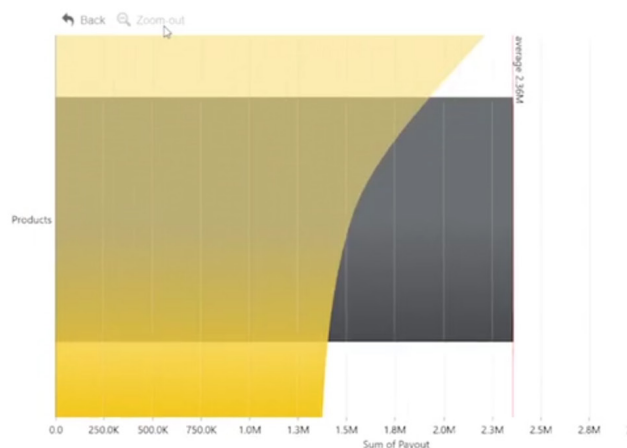
- Sales & Marketing
- Human Resources
- Accounting and Finance
- Manufacturing.



Drill Down Combo Bar PRO

Multi-level drilldown

Touch or click the category to drill down to the next level.



Paginated report visual Generally Available

The paginated report visual for Power BI is now Generally Available in this latest release of Power BI Desktop. For the first time, this native Power BI visual allows you to render any paginated report you've uploaded to Power BI Service in a Power BI report.

Charticulator (version 1.2.0)

As a reminder, Charticulator is a powerful tool for you to create custom charts for your reports using data modelled in Power BI. By representing data as glyphs and structuring the way they are repeated based on other data, you can create a diverse set of data visualisations along any permutation of mark and layout options.

This latest version of Charticulator supports a variety of new formatting features and user actions:

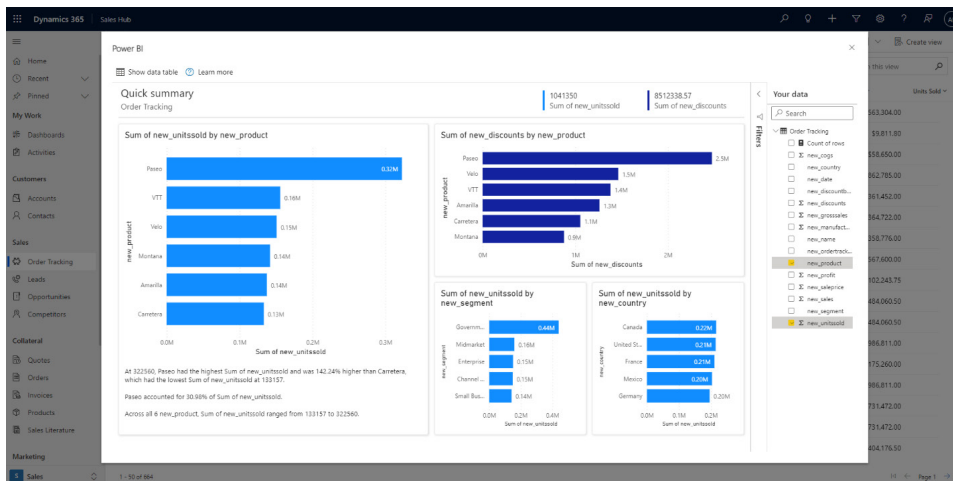
- support for exporting a template from Power BI Desktop
- support for hiding the tick line from the axis.

There are also some bugfixes and user interface (UI) improvements:

- fixed an issue with links when there is not enough data to build links
- fixed an issue with gridline rendering
- fixed the date format in the ToolTip
- moved the axis to the back in the plot segment, but made it possible to move it to the top.

Visualise your data quickly from Power Apps and Dynamics 365 apps

Microsoft recently announced a new way to visualise model-driven Power Apps and Dynamics 365 data stored within Dataverse. With a single click, Power BI automatically generates a set of visuals to explore and find insights within your data. This integration takes advantage of the full power of the Power Platform.

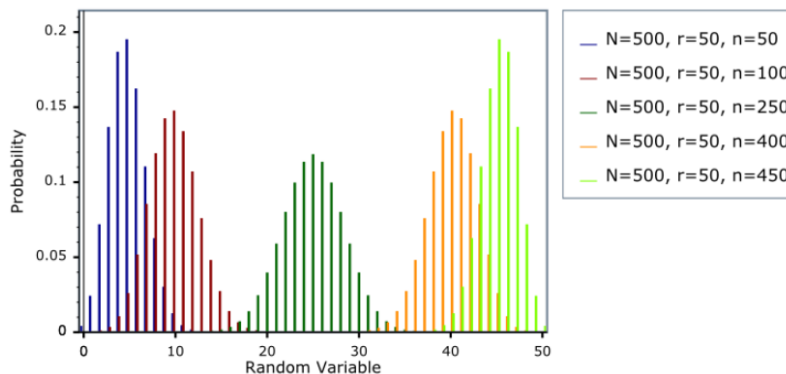


This feature is currently Previewing within Power Apps, so for the option to show, an administrator will need to enable it through the Power Apps Maker portal.

More next month, we're sure!

The A to Z of Excel Functions: HYPGEOM.DIST

Hypergeometric Distribution PDF



The hypergeometric distribution is used to calculate probabilities when sampling without replacement. For example, suppose you first randomly sample one card from a deck of 52. Then, without putting the card back in the deck you sample a second and then (again without replacing

cards) a third. Given this sampling procedure, what is the probability that exactly two of the sampled cards will be aces (four of the 52 cards in the deck are aces). You can calculate this probability using the following formula based on the hypergeometric distribution:

$$p = \frac{{}^kC_x ({}^{(N-k)}C_{(n-x)})}{{}^NC_n}$$

where:

- **k** is the number of successes in the population
- **x** is the number of successes in the sample
- **N** is the size of the population
- **n** is the number sampled
- **p** is the probability of obtaining exactly **x** successes
- ${}_kC_x$ is the number of combinations of **k** things taken **x** at a time.

In this example, **k** = 4 because there are four aces in the deck, **x** = 2 because the problem asks about the probability of getting two aces, **N** = 52 because there are 52 cards in a deck, and **n** = 3 because 3 cards were sampled. Therefore,

$$p = \frac{{}^4C_2 ({}^{(52-4)}C_{(3-2)})}{{}^{52}C_3}$$

which equates to:

$$p = \frac{\frac{4!}{2!2!} \frac{48!}{47!1!}}{\frac{52!}{49!3!}} = 0.013$$

The mean and standard deviation of the hypergeometric distribution are given by:

$$\text{mean} = \frac{(n)(k)}{N}$$

$$\text{sd} = \sqrt{\frac{(n)(k)(N-k)(N-n)}{N^2(N-1)}}$$

Note that it would not be a binomial experiment. A binomial experiment requires that the probability of success be constant on every trial. With the above experiment, the probability of a success changes on every trial. In the beginning, the probability of selecting an ace is 4/52 (=1/13). If you select an ace on the first trial, the probability of selecting an ace on the second trial is 3/51 (=1/17). If you don't select an ace on the first trial, the probability of selecting an ace on the second trial is 4/51.

Note further that if you selected the cards with replacement, the

probability of success would not change. It would be 1/13 on every trial. Then, this would be a binomial experiment.

The **HYPGEOM.DIST** function returns the hypergeometric distribution. **HYPGEOM.DIST** returns the probability of a given number of sample successes, given the sample size, population successes and population size. You should use **HYPGEOM.DIST** for problems with a finite population, where each observation is either a success or a failure, and where each subset of a given size is chosen with equal likelihood.

It has the following syntax:

HYPGEOM.DIST(sample_s, number_sample, population_s, number_population, cumulative)

The **HYPGEOM.DIST** function has the following arguments:

- **sample_s**: this is required and represents the number of successes in the sample
- **number_sample**: this is also required. This is the size of the sample
- **population_s**: again, this is required. This reflects the number of successes in the population
- **number_population**: this is required, and is the population size
- **cumulative**: this final argument is also required. This is a logical value that determines the form of the function. If **cumulative** is TRUE, then **HYPGEOM.DIST** returns the cumulative distribution function; if FALSE, it returns the probability mass function.

It should be further noted that:

- this function is a more versatile form of **HYPGEOMDIST**, which was available in older versions of Excel
- all arguments are truncated to integers
- if any argument is nonnumeric, **HYPGEOM.DIST** returns the **#VALUE!** error value
- if **sample_s** < 0 or **sample_s** is greater than the lesser of **number_sample** or **population_s**, **HYPGEOM.DIST** returns the **#NUM!** error value
- if **sample_s** is less than the larger of zero (0) or (**number_sample** - **number_population** + **population_s**), **HYPGEOM.DIST** returns the **#NUM!** error value
- if **number_sample** ≤ 0 or **number_sample** > **number_population**, **HYPGEOM.DIST** returns the **#NUM!** error value
- if **population_s** ≤ 0 or **population_s** > **number_population**, **HYPGEOM.DIST** returns the **#NUM!** error value
- if **number_population** ≤ 0, **HYPGEOM.DIST** returns the **#NUM!** error value
- as stated above, the equation for the hypergeometric distribution is:

$$P(X = x) = h(x; n, M, N) = \frac{\binom{M}{x} \binom{N - M}{n - x}}{\binom{N}{n}}$$

where:

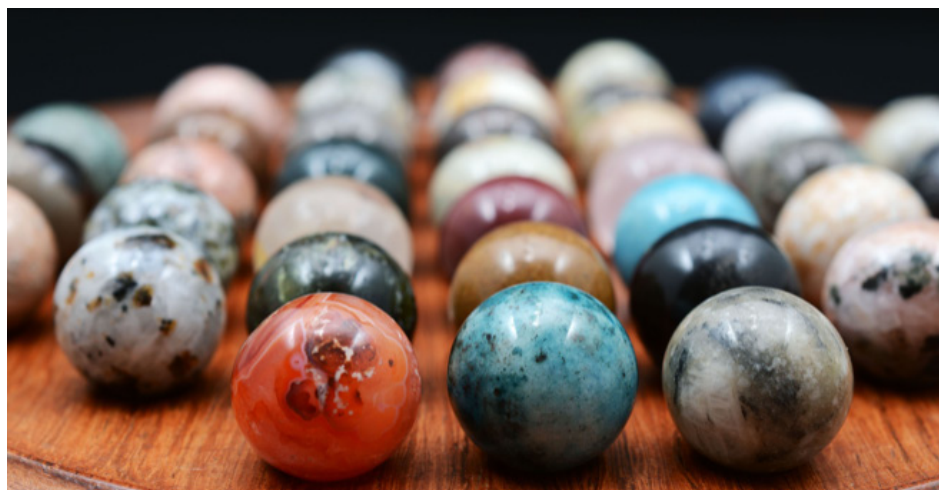
- x** = **sample_s**
- n** = **number_sample**
- M** = **population_s**
- N** = **number_population**.

HYPGEOM.DIST is used in sampling without replacement from a finite population.

Please see our example below:

	A	B	C
1	Data	Description	
2	1	Number of successes in the sample	
3	4	Sample size	
4	8	Number of successes in the population	
5	20	Population size	
6			
7			
	Formula	Description	Result
9	=HYPGEOM.DIST(A2,A3,A4,A5,TRUE)	Cumulative hypergeometric distribution function, for sample and population in cells A2:A5	0.4654
10	=HYPGEOM.DIST(A2,A3,A4,A5,FALSE)	Probability hypergeometric distribution function, for sample and in cells A2:A5	0.3633
11			

The A to Z of Excel Functions: HYPGEOMDIST



The **HYPGEOMDIST** function returns the hypergeometric distribution. **HYPGEOMDIST** returns the probability of a given number of sample successes, given the sample size, population successes and population size. You should use **HYPGEOMDIST** for problems with a finite population, where each observation is either a success or a failure, and where each subset of a given size is chosen with equal likelihood.

Erm, sound familiar..?

It has the following syntax:

HYPGEOMDIST(sample_s, number_sample, population_s, number_population)

The **HYPGEOMDIST** function has the following arguments:

- **sample_s**: this is required and represents the number of successes in the sample
- **number_sample**: this is also required. This is the size of the sample
- **population_s**: again, this is required. This reflects the number of successes in the population
- **number_population**: this final argument is also required, and is the population size.

It should be further noted that:

- this function has been replaced with **HYGEOM.DIST** (*see above!*), which may provide improved accuracy and whose names better reflect their usage. Although this function is still available for backward compatibility, you should consider using the new functions from now on, because this function may not be available in future versions of Excel
- all arguments are truncated to integers
- if any argument is nonnumeric, **HYPGEOMDIST** returns the #VALUE! error value
- if **sample_s** < 0 or **sample_s** is greater than the lesser of **number_sample** or **population_s**, **HYPGEOMDIST** returns the #NUM! error value
- if **sample_s** is less than the larger of zero (0) or (**number_sample** - **number_population** + **population_s**), **HYPGEOMDIST** returns the #NUM! error value
- if **number_sample** ≤ 0 or **number_sample** > **number_population**, **HYPGEOMDIST** returns the #NUM! error value
- if **population_s** ≤ 0 or **population_s** > **number_population**, **HYPGEOMDIST** returns the #NUM! error value
- if **number_population** ≤ 0, **HYPGEOMDIST** returns the #NUM! error value
- as stated above, the equation for the hypergeometric distribution is:

$$P(X = x) = h(x; n, M, N) = \frac{\binom{M}{x} \binom{N - M}{n - x}}{\binom{N}{n}}$$

where:

- x** = **sample_s**
- n** = **number_sample**
- M** = **population_s**
- N** = **number_population**.

HYPGEOMDIST is used in sampling without replacement from a finite population.

Please see our example below:

	A	B	C
1	Data	Description	
2	1	Number of successes in the sample	
3	4	Sample size	
4	8	Number of successes in the population	
5	20	Population size	
6			
7			
8	Formula	Description	Result
9	=HYPGEOMDIST(A2,A3,A4,A5)	Hypergeometric distribution for sample and population in cells A2:A5	0.3633
10			

More Excel Functions next month.

Beat the Boredom Suggested Solution

This month's challenge (cited earlier in the newsletter) was to convert from US date format (month – day – year) to what is known as the non-US, or more commonly, the European, date format (day – month – year). If you're in the US, the challenge was the opposite, but the issue and its solution are similar.

The problem arises when you receive date data in a spreadsheet that is not recognised by your regional settings. Therefore, for me, my computer cannot make sense of US date formats such as

	A	
1	Dates	
2	8/13/20	
3	6/14/21	
4	12/05/2022	
5	9/30/23	
6	1/29/24	
7	8/17/25	
8	5/22/26	
9	3/21/27	
10	7/14/28	
11	5/25/29	
12		

I have left the data in 'General' style deliberately so you can see only one entry, cell **A4**, is recognised as a number (date). The problem is, even that's wrong as that represents 5 December 2022 not 12 May 2022.

Get & Transform – but that's not what I want here. It may also be done with Excel's 'Text to Columns' feature, but that's not the solution here either. I want one formula to do the following:

How do I convert it? The simplest answer would be to use Power Query /

	A	B	C
1	Dates		Converted
2	8/13/20		8/13/20
3	6/14/21		6/14/21
4	12/05/2022		12/05/2022
5	9/30/23		9/30/23
6	1/29/24		1/29/24
7	8/17/25		8/17/25
8	5/22/26		5/22/26
9	3/21/27		3/21/27
10	7/14/28		7/14/28
11	5/25/29		5/25/29
12			

The Solution

As explained earlier, dates are not quite what they seem in Excel. They can cause issues for the unwary, so they are important to understand. For example, August 17, 2008 may be expressed as:

- 8/17/08 (month/day/year, US format)
- 17-8-08 (day – month - year, UK / Europe format, as it's called by Microsoft, but pretty much anywhere else)
- August 17, 2008
- 17 Aug 2008
- ...

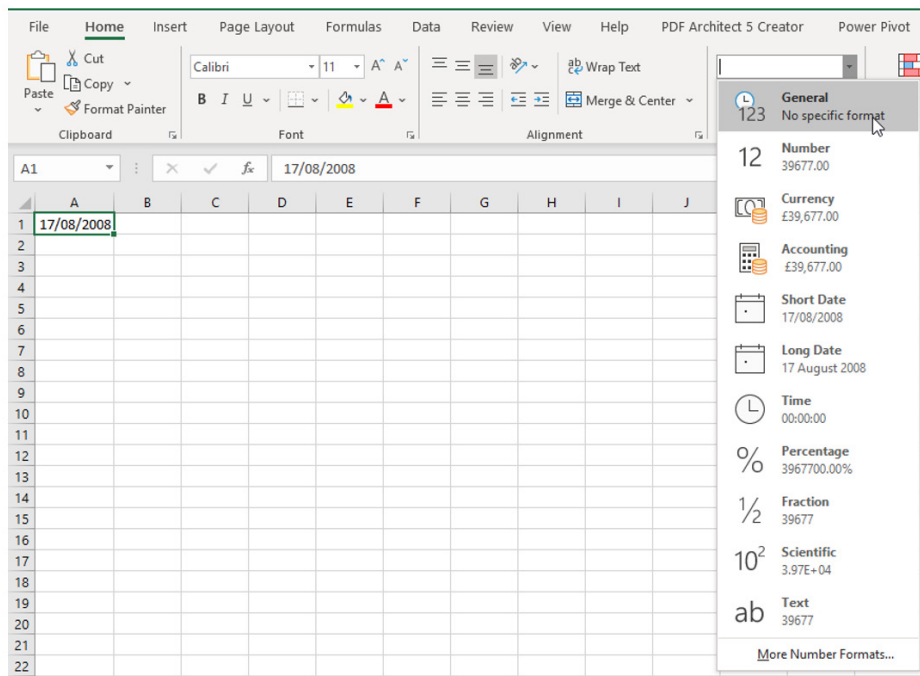
Therefore, it does not make sense for Excel to try and force a particular format and many formats are thus supported. Indeed, one thing that is more sophisticated is the fact that whether by default, day comes before

month or vice versa is down to your regional settings. This means if I type in

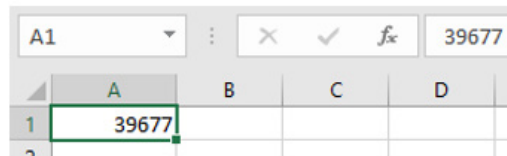
17/8/08

on some computers, this will be recognised as a date, whereas on others it will be seen as a 'General' text format because 17 exceeds the number of months in a year. For our American readers, I apologise in advance: my computer is not set to US settings and therefore, the examples here will be displayed as day followed by month followed by year. In the US, the month preceded the day.

The point is these are display issues only. As a non-US Excel user, I will type "17/8/08" into cell **A1**, and then format the cell as 'General' (using the drop-down box in the 'Number' grouping of the 'Home' tab):



This changes cell A1 as follows:

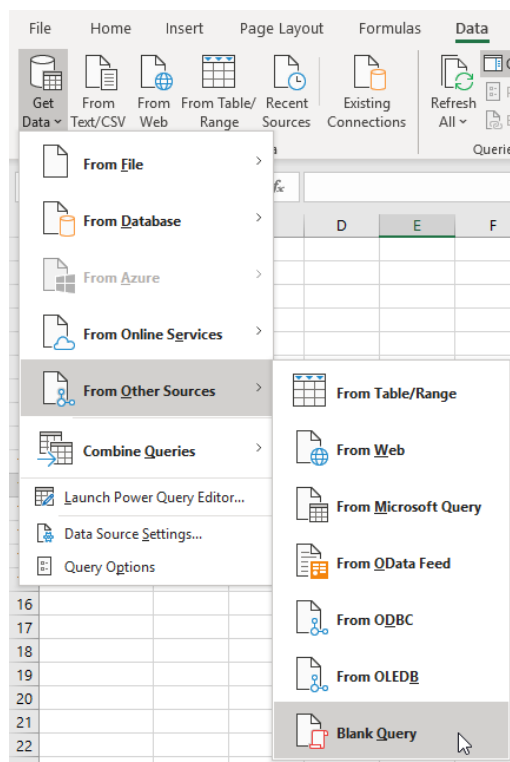


It becomes “39677”, which is known as a **serial number** (if I were to delete this value, it would make me a serial killer). This serial number is a count from “Day 1” which is defined as 1 January 1900, so Day 2 would be 2 January 1900, Day 3 would be 3 January 1900, and so on. Therefore, according to Excel, 17 August 2008 is Day 39,677.

In essence, we just need Excel to recognise the date as a date – and that was the issue here.

Just to be clear (before everyone writes in!), you can solve this using Power Query / Get & Transform by applying a series of transformations (sometimes it’s just a matter of changing locales, but not always).

For example, if the dates were in an Excel Table (using **CTRL + T**, called **Original_Dates**), you could use ‘Get & Transform’ / Power Query by creating a Blank Query,



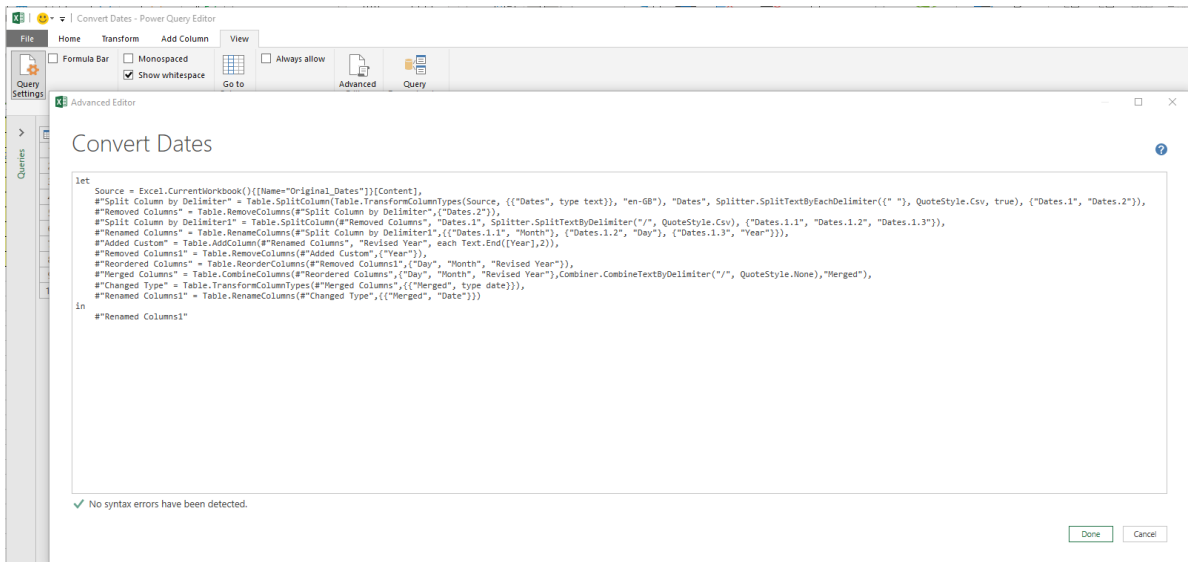
and pasting the following code into the Advanced Editor:

let

```
Source = Excel.CurrentWorkbook(){[Name="Original_Dates"]}[Content],
#"Split Column by Delimiter" =
Table.SplitColumn(Table.TransformColumnTypes(Source, {"Dates", type text}), "en-GB", "Dates",
Splitter.SplitTextByEachDelimiter({" "}, QuoteStyle.Csv, true), {"Dates.1", "Dates.2"}),
#"Removed Columns" = Table.RemoveColumns("#Split Column by Delimiter",{"Dates.2"}),
#"Split Column by Delimiter1" = Table.SplitColumn("#Removed Columns", "Dates.1", Splitter.
SplitTextByDelimiter("/", QuoteStyle.Csv), {"Dates.1.1", "Dates.1.2", "Dates.1.3"}),
#"Renamed Columns" = Table.RenameColumns("#Split Column by Delimiter1",{"Dates.1.1", "Month"},
{"Dates.1.2", "Day"}, {"Dates.1.3", "Year"}),
#"Added Custom" = Table.AddColumn("#Renamed Columns", "Revised Year", each Text.End([Year],2)),
#"Removed Columns1" = Table.RemoveColumns("#Added Custom",{"Year"}),
#"Reordered Columns" = Table.ReorderColumns("#Removed Columns1",{"Day", "Month", "Revised
Year"}),
#"Merged Columns" = Table.CombineColumns("#Reordered Columns",{"Day", "Month", "Revised
Year"},Combiner.CombineTextByDelimiter("/", QuoteStyle.None),"Merged"),
#"Changed Type" = Table.TransformColumnTypes("#Merged Columns",{"Merged", type date}),
#"Renamed Columns1" = Table.RenameColumns("#Changed Type",{"Merged", "Date"})
```

in

```
#"Renamed Columns1"
```

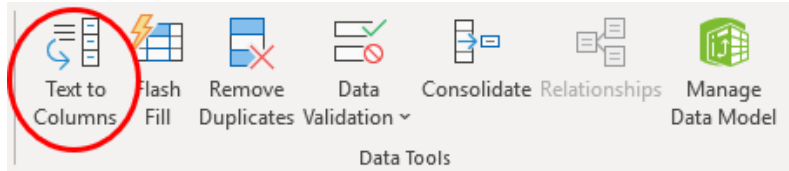


You could also do it in good old Excel by using the 'Text to Columns' method as follows. First, I make a copy in cells C2:C11:

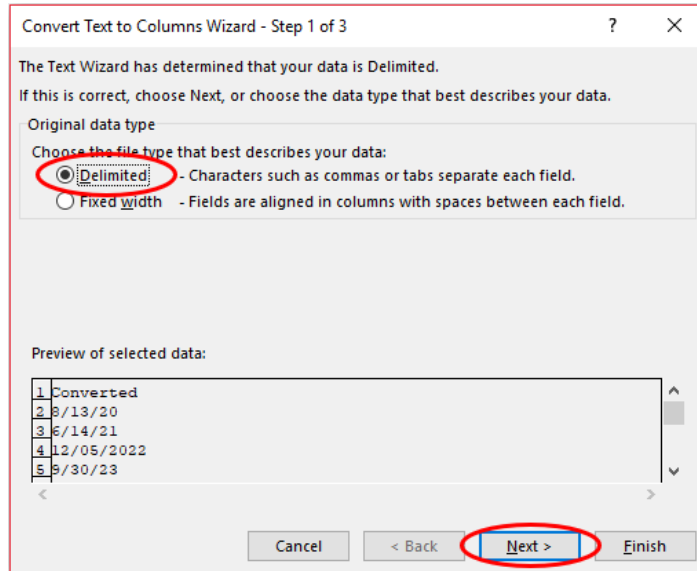
	A	B	C
1	Dates		Converted
2	8/13/20		8/13/20
3	6/14/21		6/14/21
4	12/05/2022		12/05/2022
5	9/30/23		9/30/23
6	1/29/24		1/29/24
7	8/17/25		8/17/25
8	5/22/26		5/22/26
9	3/21/27		3/21/27
10	7/14/28		7/14/28
11	5/25/29		5/25/29

I do this so I may retain the original data (it's always best to keep a copy in case you make a mistake).

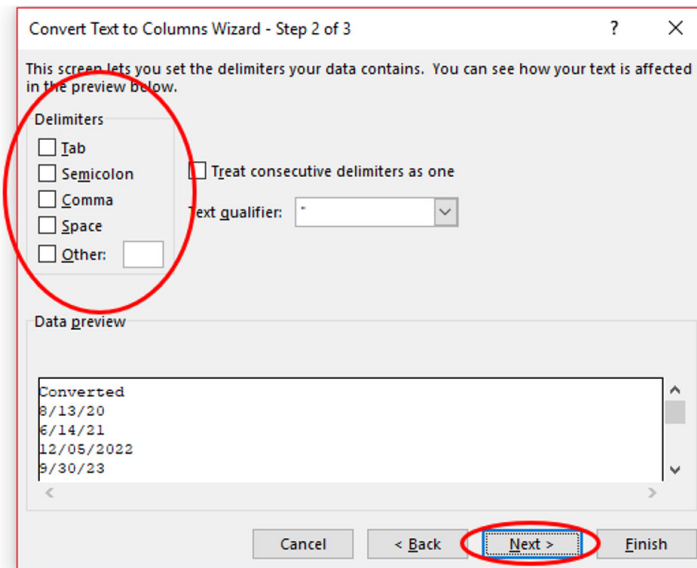
Next, I highlight cells **C1:C11** (including the header) and click on 'Text to Columns' in the 'Data Tools' grouping of the 'Data' tab of the ribbon (**ALT + A + E**):



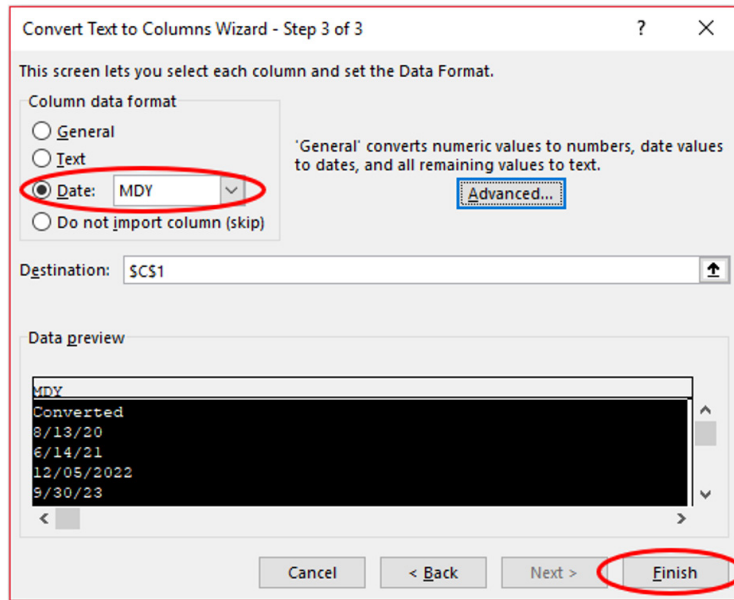
This generates the 'Convert Text to Columns Wizard' dialog box. In 'Step 1', choose the 'Delimited' option and click 'Next'.



This means the data will be split into columns based upon a specified delimiter. Except we are going to cheat and not do that. In 'Step 2', uncheck all delimiters and then click 'Next':



Now we come to the step that we actually want. We don't use the 'Text to Columns' feature to split data into separate columns. No, I want Excel to recognise my data as dates.



In this final step, select the 'Date:' option in the 'Column data format' and choose the date format that matches the data *as it currently is* – not what you want it to be. You are asking Excel to recognize it. In my case, the data is in Month – Day – Year format (MDY), so this is what I selected. Once you have chosen, click 'Finish':

	A	B	C
1	Dates		Converted
2	8/13/20		13/08/2020
3	6/14/21		14/06/2021
4	12/05/2022		05/12/2022
5	9/30/23		30/09/2023
6	1/29/24		29/01/2024
7	8/17/25		17/08/2025
8	5/22/26		22/05/2026
9	3/21/27		21/03/2027
10	7/14/28		14/07/2028
11	5/25/29		25/05/2029

This is arguably the simplest method.

However, both of these methods require data to be refreshed. The challenge was to construct a formulaic approach. I must warn you, it's not pretty, but then, neither am I. The point is sometimes brute force and ignorance get the job done. Here's my approach in full.

To begin, I convert the date as follows:

	A	B	C	D	E
1	Dates		Converted Date		
2	8/13/20		8@13/20		=SUBSTITUTE(IFERROR(TEXT(A2,"d/m/yy"),A2),"/","@"&".1)
3	6/14/21		6@14/21		
4	12/05/2022		12@5/22		
5	9/30/23		9@30/23		
6	1/29/24		1@29/24		
7	8/17/25		8@17/25		
8	5/22/26		5@22/26		
9	3/21/27		3@21/27		
10	7/14/28		7@14/28		
11	5/25/29		5@25/29		

The formula

`=TEXT(A2,"d/m/yy")`

converts cell **A2** to a text string in the format day-month-year (which is what you want the data converted to, not the source). The reason for this is if you used the format code `"m/d/yy"` instead, cell A4 would have the day and month swapped around incorrectly (as it is already recognised by Excel as a date which is why this value is right-aligned).

The wraparound

`=IFERROR(TEXT(A2,"d/m/yy"),A2)`

is added because for certain regional settings where **A2** was originally recognised as a date (like the value in cell **A4** here where 5-Dec-22 has been mistaken for 12-May-22), this may cause an error as it is already recognised as a date string. In this instance, the original value is used instead.

Finally, the formula

`=SUBSTITUTE(IFERROR(TEXT(A2,"d/m/yy"),A2),"/","@",1)`

uses the **SUBSTITUTE** function

`SUBSTITUTE(text,old_text,new_text,[instance_number])`

to replace one or more instances of a given character or text string (**old_text**) in a text string with a specified character or string (**new_text**). The optional **instance_number** cites the occurrence of **old_text** you wish to replace. If this is omitted, all occurrences will be replaced.

Here, I have replaced the first `"/"` with `"@"`. The choice of `"@"` is arbitrary: I am trying to get the values between the `"/"` delimiters and having two of these is awkward, so I have replaced the `"/"` with `"@"` because I know it's a character I won't see in a date text string!

Next, I find the position of the `"@"` symbol in the text string just created in column C:

	A	B	C	D	E	F	G
1	Dates		Converted Date		Find @		
2	8/13/20		8@13/20		2		=FIND("@",C2)
3	6/14/21		6@14/21		2		
4	12/05/2022		12@5/22		3		
5	9/30/23		9@30/23		2		
6	1/29/24		1@29/24		2		
7	8/17/25		8@17/25		2		
8	5/22/26		5@22/26		2		
9	3/21/27		3@21/27		2		
10	7/14/28		7@14/28		2		
11	5/25/29		5@25/29		2		

`FIND(find_text,within_text,[start_number])`

is a search function which is case sensitive but does not allow wildcard characters. It seeks out the first instance of a character or characters (typed in inverted commas) in the **within_text** text string. The **start_number** argument is optional (hence the square brackets in the syntax) so that the first few characters in a text string may be ignored. If the **find_text** cannot be located within **within_text** the error `#VALUE!` is returned.

Here, the formula

`=FIND("@",C2)`

locates the position of `"@"` in each text string (2 for all but `"12/05/2022"`). This is important as we know the first set of numbers (month number) precedes this character and the second set (day number) immediately follows it.

Now we locate `"/"`:

	A	B	C	D	E	F	G	H	I
1	Dates		Converted Date		Find @		Find /		
2	8/13/20		8@13/20		2		5		=FIND("/",C2)
3	6/14/21		6@14/21		2		5		
4	12/05/2022		12@5/22		3		5		
5	9/30/23		9@30/23		2		5		
6	1/29/24		1@29/24		2		5		
7	8/17/25		8@17/25		2		5		
8	5/22/26		5@22/26		2		5		
9	3/21/27		3@21/27		2		5		
10	7/14/28		7@14/28		2		5		
11	5/25/29		5@25/29		2		5		

This is similar to the last formula:

=FIND("/",C2)

This locates the position of "/" (we aren't having to hunt for a second one) in each text string. This is important as we know the numbers before represent the day number and those immediately following represent the year.

We are now in a position to extract the day number:

	A	B	C	D	E	F	G	H	I	J	K
1	Dates		Converted Date		Find @		Find /		Day		
2	8/13/20		8@13/20		2		5		13		=MID(C2,E2+1,G2-E2-1)
3	6/14/21		6@14/21		2		5		14		
4	12/05/2022		12@5/22		3		5		5		
5	9/30/23		9@30/23		2		5		30		
6	1/29/24		1@29/24		2		5		29		
7	8/17/25		8@17/25		2		5		17		
8	5/22/26		5@22/26		2		5		22		
9	3/21/27		3@21/27		2		5		21		
10	7/14/28		7@14/28		2		5		14		
11	5/25/29		5@25/29		2		5		25		

The formula

=MID(C2,E2+1,G2-E2-1)

relies upon the **MID** function, which is short for "Middle", which means you can capture parts of a word, starting at any point in a cell.

Its syntax is as follows:

=MID(text, start_number, number_of_characters).

with the following arguments:

- **text** is as before and refers to the cell / string you wish to convert
- **start_number** is the starting position (that is, which will be the first character in the text string subset to be extracted). For example, 2 would mean the second character, 3 the third character and so on
- **number_of_characters** is the length of the extracted subset text string.

In our formula,

=MID(C2,E2+1,G2-E2-1)

C2 – not **A2**, which would not work where the value is recognised as a date and has been converted to a serial number – is used as the **text**. The start number is the position immediately after the "@" symbol (denoted by **E2+1**) and the number of characters is the number between the "@" and the "/" symbols, which is given by **G2-E2-1** in row 2.

The month number is easier:

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Dates		Converted Date		Find @		Find /		Day		Month		
2	8/13/20		8@13/20		2		5		13		8		=LEFT(C2,E2-1)
3	6/14/21		6@14/21		2		5		14		6		
4	12/05/2022		12@5/22		3		5		5		12		
5	9/30/23		9@30/23		2		5		30		9		
6	1/29/24		1@29/24		2		5		29		1		
7	8/17/25		8@17/25		2		5		17		8		
8	5/22/26		5@22/26		2		5		22		5		
9	3/21/27		3@21/27		2		5		21		3		
10	7/14/28		7@14/28		2		5		14		7		
11	5/25/29		5@25/29		2		5		25		5		

This uses the **LEFT** function:

=LEFT(text, [number_of_characters])

The **text** argument refers to the cell you want to convert and **[number_of_characters]** is the number of characters you want to extract from the cell, counting from the **left**. The reason the argument is optional (that is, there are square brackets around the second argument in the syntax) is because if it is not specified it is assumed to be one [1].

Here, the formula in cell **K2** is given by

=LEFT(C2,E2-1)

where again we refer to **C2** as the **text** (for the same reasons as before). The number of characters is **E2-1**, *i.e.* all of the characters before the “@” symbol.

The year is just the last two digits of the text string in column **C**:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Dates		Converted Date	Find @	Find /	Day	Month	Year							
2	8/13/20		8@13/20	2	5	13	8	20							
3	6/14/21		6@14/21	2	5	14	6	21							
4	12/05/2022		12@5/22	3	5	5	12	22							
5	9/30/23		9@30/23	2	5	30	9	23							
6	1/29/24		1@29/24	2	5	29	1	24							
7	8/17/25		8@17/25	2	5	17	8	25							
8	5/22/26		5@22/26	2	5	22	5	26							
9	3/21/27		3@21/27	2	5	21	3	27							
10	7/14/28		7@14/28	2	5	14	7	28							
11	5/25/29		5@25/29	2	5	25	5	29							

This uses the **RIGHT** function, which works similarly to **LEFT**, except it starts from the **right** and counts backwards. Everything else is the same. Therefore,

=RIGHT(C2,2)

extracts the last two characters of the **text** string in cell **C2**.

We just have to put the date together now:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Dates		Converted Date	Find @	Find /	Day	Month	Year							Revised Date		
2	8/13/20		8@13/20	2	5	13	8	20							13/08/2020		
3	6/14/21		6@14/21	2	5	14	6	21							14/06/2021		
4	12/05/2022		12@5/22	3	5	5	12	22							05/12/2022		
5	9/30/23		9@30/23	2	5	30	9	23							30/09/2023		
6	1/29/24		1@29/24	2	5	29	1	24							29/01/2024		
7	8/17/25		8@17/25	2	5	17	8	25							17/08/2025		
8	5/22/26		5@22/26	2	5	22	5	26							22/05/2026		
9	3/21/27		3@21/27	2	5	21	3	27							21/03/2027		
10	7/14/28		7@14/28	2	5	14	7	28							14/07/2028		
11	5/25/29		5@25/29	2	5	25	5	29							25/05/2029		

The formula

=I2&"/"&K2&"/"&M2

joins the day, month and year up using “/” delimiters. However, this is by definition a text string. Therefore, the actual formula in cell **O2**,

=VALUE(I2&"/"&K2&"/"&M2)

then converts this to a (serial) number. Column **O** has then been formatted as a short date and the Revised Date has finally been created.

The problem is, we said we wanted **one** formula. This means **I2**, **K2** and **M2** have to be replaced by their dependent formulae in cell **O2**, and then these formulae need to keep referencing back too, until all transformations refer back to column **A**. When you do this, you get the following wondrous megaformula:

	A	B	C	D	E
1	Dates		Revised Date		
2	8/13/20		13/08/2020		=VALUE(MID(SUBSTITUTE(IFERROR(TEXT(A2,"d/m/y
3	6/14/21		14/06/2021		y"),A2),"/","@",1),FIND("@",SUBSTITUTE(IFERROR(T
4	12/05/2022		05/12/2022		EXT(A2,"d/m/yy"),A2),"/","@",1))+1,FIND("/","SUBST
5	9/30/23		30/09/2023		ITUTE(IFERROR(TEXT(A2,"d/m/yy"),A2),"/","@",1))-
6	1/29/24		29/01/2024		FIND("@",SUBSTITUTE(IFERROR(TEXT(A2,"d/m/yy"),
7	8/17/25		17/08/2025		A2),"/","@",1))-
8	5/22/26		22/05/2026		1)&"/"&LEFT(SUBSTITUTE(IFERROR(TEXT(A2,"d/m/y
9	3/21/27		21/03/2027		y"),A2),"/","@",1),FIND("@",SUBSTITUTE(IFERROR(T
10	7/14/28		14/07/2028		EXT(A2,"d/m/yy"),A2),"/","@",1))-
11	5/25/29		25/05/2029		1)&"/"&RIGHT(SUBSTITUTE(IFERROR(TEXT(A2,"d/m/
12					yy"),A2),"/","@",1,2))

```
=VALUE(MID(SUBSTITUTE(IFERROR(TEXT(A2,"d/m/yy"),A2),"/","@",1),
FIND("@",SUBSTITUTE(IFERROR(TEXT(A2,"d/m/yy"),A2),"/","@",1))+1,
FIND("/","SUBSTITUTE(IFERROR(TEXT(A2,"d/m/yy"),A2),"/","@",1))-
FIND("@",SUBSTITUTE(IFERROR(TEXT(A2,"d/m/yy"),A2),"/","@",1))-1
&"/"&LEFT(SUBSTITUTE(IFERROR(TEXT(A2,"d/m/yy"),A2),"/","@",1),
FIND("@",SUBSTITUTE(IFERROR(TEXT(A2,"d/m/yy"),A2),"/","@",1))-1
&"/"&RIGHT(SUBSTITUTE(IFERROR(TEXT(A2,"d/m/yy"),A2),"/","@",1),2))
```

Beautiful.

Until next time.

Upcoming SumProduct Training Courses - COVID-19 update

Due to the COVID-19 pandemic that is currently spreading around the globe, we are suspending our in-person courses until further notice. However, to accommodate the new working-from-home dynamic, we are switching our public and in-house courses to an online delivery stream, presented via Microsoft Teams, with a live presenter running through the same course material, downloadable workbooks to complete the hands-on exercises during the training session, and a recording of the sessions for

your use within 1 month for you to refer back to in the event of technical difficulties. To assist with the pacing and flow of the course, we will also have a moderator who will help answer questions during the course.

If you're still not sure how this will work, please contact us at training@sumproduct.com and we'll be happy to walk you through the process.

Location	Course	Date	Date	Duration	Duration
Online (Australia)	Power Pivot, Power Query and Power BI	16 - 18 Feb 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	23 Feb 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	1 Day
Online (Australia)	Financial Modelling	24 - 25 Feb 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	2 Days
Online (Australia)	Excel Tips and Tricks	11 Apr 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	1 Day
Online (Australia)	Financial Modelling	12 - 13 Apr 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	10 - 12 May 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	17 May 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	1 Day
Online (Australia)	Financial Modelling	18 - 19 May 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	19 - 21 Jul 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	26 Jul 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	1 Day
Online (Australia)	Financial Modelling	27 - 28 Jul 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	2 Days
Online (Australia)	Excel Tips and Tricks	29 Aug 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	1 Day
Online (Australia)	Financial Modelling	30 - 31 Aug 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	28 -30 Sep 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	5 Oct 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	1 Day
Online (Australia)	Financial Modelling	6 - 7 Oct 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	2 Days

Location	Course	Date	Date	Duration	Duration
Online (Australia)	Power Pivot, Power Query and Power BI	9 - 11 Nov 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	16 Nov 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	1 Day
Online (Australia)	Financial Modelling	17 - 18 Nov 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	7 - 9 Dec 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	14 Dec 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	1 Day
Online (Australia)	Financial Modelling	15 - 16 Dec 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	2 Days

Key Strokes

Each newsletter, we'd like to introduce you to useful keystrokes you may or may not be aware of. This month we take **CTRL** of the numbers:

Keystroke	What it does
CTRL + 0	Hide column
CTRL + 1	Format cells
CTRL + 2	Bold (toggle)
CTRL + 3	Italicise (toggle)
CTRL + 4	Underline (toggle)
CTRL + 5	Strikethrough (toggle)
CTRL + 6	Toggle Show, Hide and Show Placeholders for Objects
CTRL + 7	Excel 2003: toggle Standard toolbar visibility
CTRL + 8	Toggle showing outline symbols
CTRL + 9	Hide row

There are c.550 keyboard shortcuts in Excel. For a comprehensive list, please download our Excel file at www.sumproduct.com/thought/keyboard-shortcuts. Also, check out our new daily **Excel Tip of the Day** feature on the www.sumproduct.com homepage.

Our Services

We have undertaken a vast array of assignments over the years, including:

- **Business planning**
- **Building three-way integrated financial statement projections**
- **Independent expert reviews**
- **Key driver analysis**
- **Model reviews / audits for internal and external purposes**
- **M&A work**
- **Model scoping**
- **Power BI, Power Query & Power Pivot**
- **Project finance**
- **Real options analysis**
- **Refinancing / restructuring**
- **Strategic modelling**
- **Valuations**
- **Working capital management**

If you require modelling assistance of any kind, please do not hesitate to contact us at contact@sumproduct.com.

Link to Others

These newsletters are not intended to be closely guarded secrets. Please feel free to forward this newsletter to anyone you think might be interested in converting to "the SumProduct way".

If you have received a forwarded newsletter and would like to receive future editions automatically, please subscribe by completing our newsletter registration process found at the foot of any www.sumproduct.com web page.

Any Questions?

If you have any tips, comments or queries for future newsletters, we'd be delighted to hear from you. Please drop us a line at newsletter@sumproduct.com.

Training

SumProduct offers a wide range of training courses, aimed at finance professionals and budding Excel experts. Courses include Excel Tricks & Tips, Financial Modelling 101, Introduction to Forecasting and M&A Modelling.

Check out our more popular courses in our training brochure:



Drop us a line at training@sumproduct.com for a copy of the brochure or download it directly from www.sumproduct.com/training.