

# Sum Product

NEWSLETTER #108 - November 2021

[www.sumproduct.com](http://www.sumproduct.com) | [www.sumproduct.com/thought](http://www.sumproduct.com/thought)



*As we get towards the end of the year* the SumProduct newsletter train rolls on and on. We look back at last month's Excel Virtually Global, look forward to Excel charging into space, report on Recommended Files and the further demise of Excel 4 macros, plus highlight the improvements being made to scrolling in Excel.

And then there is all the usual stuff: with another Beat the Boredom Challenge, plus Charts & Dashboards, Visual Basics, Power Pivot Principles, Power Query Pointers, Power BI updates, Keyboard Shortcuts and we mean no **HARMEAN** with this month's A to Z of Excel Functions.

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

*Liam Bastick*, Managing Director, SumProduct



## Excel Virtually Global This Month - Been and Gone!



# EXCEL VIRTUALLY GLOBAL 2021

We came, we saw, we brought the popcorn.

Excel Virtually Global, our annual Excel / Power BI event now being held for its seventh successive year in its different guises, has been and gone. With over 50 presenters, we learned a lot about Power Pivot, Power Query, Power BI, PowerPoint, Excel and financial modelling.

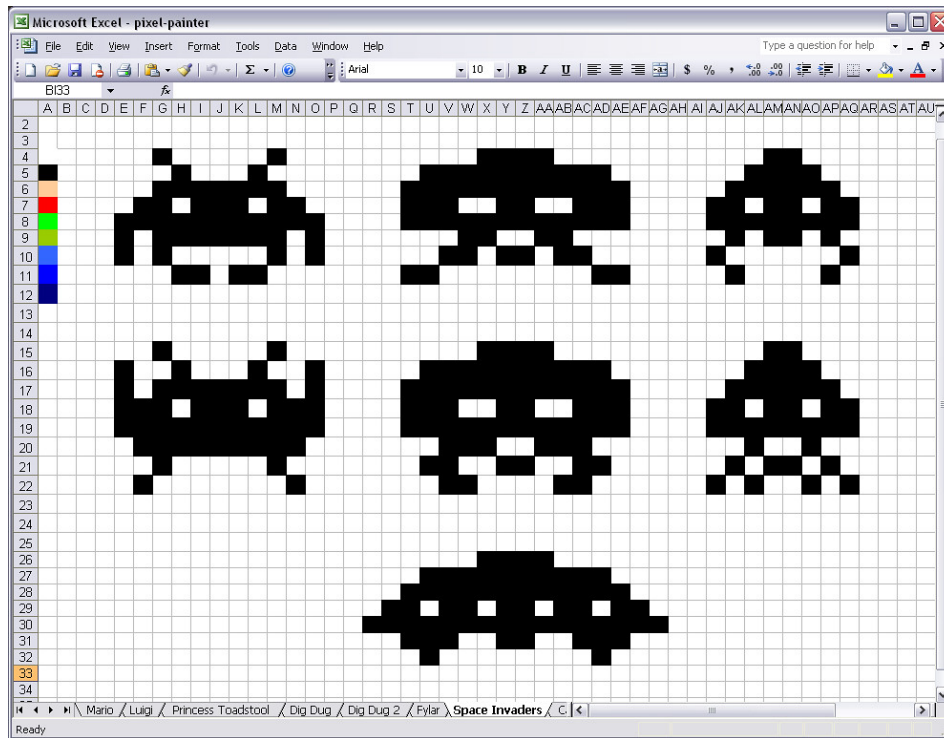
We had over 50 wonderful presenters and an engaged audience. If you missed it, it's still possible to watch many of the sessions. Simply go to [www.sumproduct.com/excel-virtually-global-2021](http://www.sumproduct.com/excel-virtually-global-2021) for further details.

Please don't forget, this event was for charity. We'd love you to donate to a worthy cause of your choice and just let us know by filling in the form using the QR code →

or else use the link [bit.ly/EVG2021Charity](https://bit.ly/EVG2021Charity) (this is case sensitive and should be typed in "as is").



## Excel in Space



It's nice to see Excel in the news for exploratory reasons.

Recently, NASA's latest exploration spacecraft launched, "The Lucy mission", beginning a 12-year journey to visit Jupiter's Trojan asteroids. The Lucy spacecraft is expected to travel further than four billion miles (over six billion kilometres) on a "fly by" mission to study eight of Jupiter's Trojan asteroids.

Needing to consider speeds through near vacuums, the impact of dust particles on trajectory, solar winds, speeds and orbits of heavenly bodies,

requires complex calculations using some of the most sophisticated software on the planet. So what did the boffins use? Why, Excel of course!

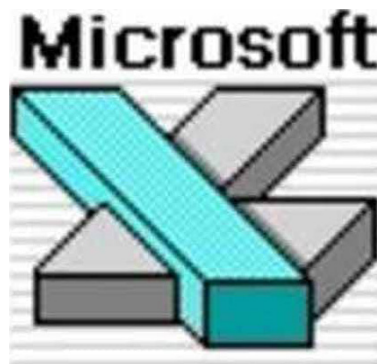
The mission's architect, Brian Sutter, used Excel and its macros to chart the mission's path and choose which of the about 5,000 Trojan asteroids the spacecraft should visit. Amazing.

We hope he remembered to carry the one and ensure Excel's calculate settings were not set to 'Manual'... I love Lucy loves Excel.

## Microsoft to Disable Excel 4 Macros

Microsoft will soon disable Excel 4.0 XLM macros by *default*. This is because it's been recognised that these types of macros may be used to get malware onto people's PCs. Attackers can place XLM macros into

malicious documents that download malware onto the computers of unsuspecting victims. The switch will disable these types of macros by default in Microsoft 365 tenants.



Microsoft recommends that people use VBA macros or Office Scripts instead. VBA macros support Antimalware Scan Interface (AMSI), which scans documents for malware and other dangerous content.

Windows administrators may disable XLM macros through the Excel

Trust Center, although this will soon be unnecessary. Preview builds will have XLM macros disabled by default shortly (if not by the time you are reading this), and the change will roll out to the Current Channel in November.

# Smooth Scrolling in Excel

Ever found it difficult to scroll through a worksheet with large cells and view all the data? According to Microsoft, the main reason for this issue is that Excel automatically snaps to the top-left cell as you scroll. Although you might think this is an easy fix, simply requiring the alteration of a few lines of code, apparently there's actually a lot more to it. In fact, the changes required affect many different aspects of Excel, including freezing panes, resizing rows, cutting and pasting, filtering, cell styles, comments, dragging and filling, and more.

It should be noted that depending upon the mouse or touch pad you're using, you may not be able to scroll one pixel at a time. If you don't have a "precision" mouse or touch pad, each movement will be based on the height of one line of text. If your Windows setting is to move one line for each click of the scroll wheel, it will move by line of text rather than by one row of your worksheet.

When you're scrolling in a sheet with very large cells, the way Excel for Windows snapped to the grid made it hard to view your data. Now, with this update, the scrolling is much smoother.

There are two aspects to this update:

1. It's smoother during the scroll when using either the mouse wheel or the scrollbars (touch screen and touch pad already scrolled smoothly)
2. You can stop scrolling partway through a row or column, and Excel won't force you to go any further than you want.

Microsoft Excel team staff member Steve Kravak provided some interesting insights into the story of this improvement. He commented that it provided an interesting peek into the complexities of making even small changes to Excel, which has been around for decades and has so many features. Many who have asked for Excel to stop snapping when you're trying to scroll just a little have also asked why this wasn't done a

long time ago. After all, it should be a simple matter of removing a few lines of code that causes the sheet to snap. Apparently, it's not that simple!

Apparently, two main types of issues as a result of not snapping to the edges and scrolling smoothly were identified:

1. Issues dealing with the positioning of objects on the sheet
2. Issues relating to how objects are displayed when scrolling.

To understand why there were issues with positioning objects on the sheet, it's helpful to know that the position of the sheet (which cell is in the top-left corner of the visible area) is fundamental to just about everything that gets displayed in Excel. For example, if you have a chart, Excel calculates the position where the chart should be displayed based on the distance from the top left cell.

Calculations like this were created with the understanding that the sheet would always be positioned cleanly on the edge of a row and column. Now that smooth scrolling allows the sheet to be positioned partway through rows and columns, those calculations had to be updated to avoid your chart showing up in the wrong place. Likewise, there are calculations to determine where objects and screen elements should appear while scrolling. Many of these calculations assumed that Excel would always be cleanly snapped to cell edges. To provide a smooth experience, the Excel team had to come up with new ways to calculate these positions in a more dynamic way.

They also uncovered issues related to the location of mouse and touch points in the sheet. There are calculations to determine where you're clicking or pointing in Excel. If these calculations assume the sheet would be snapped, your click might not hit what you expect it to hit. For example, you may click on a cell, but a different cell would be selected.

	B	C	D	E	F	G	H
1		Product	Sales	Rating			
2	nts	Chains	\$ 20,000	75%			
3		Socks	\$ 3,700	22%			
4		Bib-Shorts	\$ 4,000	22%			
5		Shorts	\$ 13,300	56%			
6		Tights	\$ 36,000	100%			
7	nts	Handlebars	\$ 2,300	35%			
8		Socks	\$ 2,300	28%			
9	nts	Brakes	\$ 3,400	36%			
10		Mountain Bikes	\$ 6,300	40%			
11	nts	Brakes	\$ 5,400	38%			

The above image shows the mouse is clicking on cell E5, but cell G5 becomes selected, due to an offset from column B being scrolled part way out of view, causing an "offset", which was not possible when Excel always snapped to the edge of a column.

Using a touchscreen has similar problems. Excel calculates what should be in the place where you touched, and it always assumed the sheet would be snapped. One issue found was that if you scrolled partway through a row and tried to draw on the sheet, there would be a gap near the bottom of the sheet where Excel couldn't draw anything. The gap was exactly the same height as the part of the top row that was scrolled out of view.

	A	B	C	D	E
1		Contoso Bike Sales product, and rating.			
2	Year	Category	Product	Sales	Rating
3	2017	Components	Chains	\$ 20,000	75%
4	2015	Clothing	Socks	\$ 3,700	22%
5	2017	Clothing	Bib-Shorts	\$ 4,000	22%
6	2015	Clothing	Shorts	\$ 13,300	56%
7	2017	Clothing	Tights	\$ 36,000	100%
8	2015	Components	Handlebars	\$ 2,300	35%
9	2016	Clothing	Socks	\$ 2,300	28%
10	2016	Components	Brakes	\$ 3,400	36%
11	2016	Bikes	Mountain Bikes	\$ 6,300	40%
12	2017	Components	Brakes	\$ 5,400	38%
13	2016	Accessories	Helmets	\$ 17,000	90%
14	2016	Accessories	Bottom Brackets	\$ 21,600	90%
15	2016	Accessories	Road Bikes	\$ 29,800	90%
16	2016	Components	Jerseys	\$ 1,000	23%
17	2015	Clothing	Tires and Tubes	\$ 6,700	5%
18	2017	Components	Bottom Brackets	\$ 600	27%
19	2015	Bikes	Road Bikes	\$ 3,500	50%
20	2017	Clothing	Jerseys	\$ 7,500	40%
21	2017	Accessories	Tires and Tubes	\$ 63,700	90%
22	2017	Bikes	Cargo Bike	\$ 9,300	60%
23	2017	Bikes	Mountain Bikes	\$ 8,500	46%

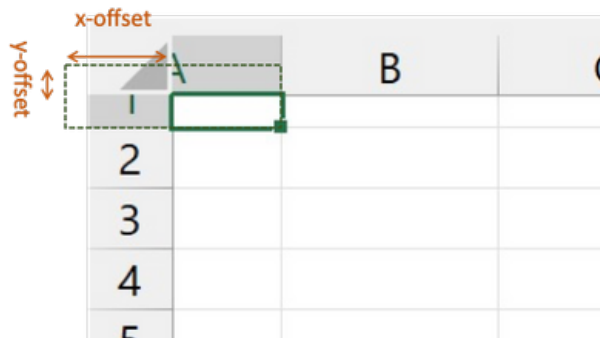
This image above shows that since row 1 is scrolled partway out of view, the “offset” caused a drawing problem near the bottom of the sheet. Drawing near the bottom of the sheet stopped working in an area the same height as the amount of row 1 which is out of view.

Microsoft even came up with a word cloud graphic to list various contributing factors!



The diagram below shows that with smooth scrolling, the sheet can be scrolled partway through a row and column, which creates an “offset” distance both horizontally and vertically. This offset had never been possible since the sheet was always snapped to the edge of a row and

column. The calculations of where things are positioned on the screen was straightforward. Now the calculations need to deal with the offset, so it’s significantly more complicated.



Microsoft found in early testing that some things just didn’t look or work as expected, the Excel team had to take a step back and design a fix in a new way that would avoid causing additional problems.

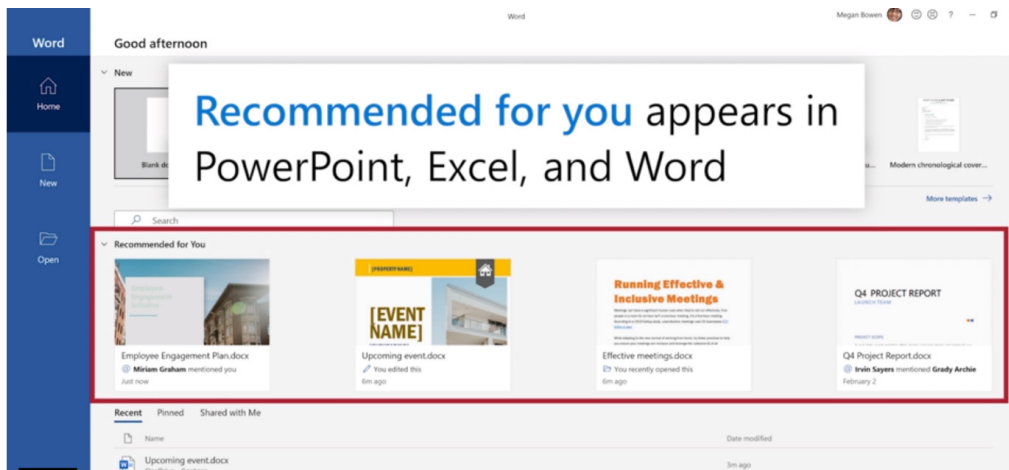
Another consideration was the variety of different ways that you can scroll in Excel, e.g. mouse wheel, touch pad, touch screen, scroll bar drag, scroll bar buttons and selection drag. Each scroll method has differences that could be impacted by these changes. The team spent time working to streamline the code for each of the scrolling methods to reduce the

differences. This means that fixing an issue for one scrolling method would likely fix it for all. It was time well spent, but it delayed the release of Smooth Scrolling!

This improvement is available to Insiders on Windows with Beta Channel or Current Channel Preview with Version 2109 (Build 14430.20000) or later. Features will be enabled gradually over some time to ensure things are working properly. Therefore, it should be available to all M365 Subscribers in the coming months.

## Recommended Files

Microsoft Office will soon display a list of recommended files on the File tab or start page of Word, Excel or PowerPoint on Windows. This list allows you to keep track of work around you and quickly access files with activity you value most such as edits, mentions and comments by people you interact with.



This feature uses machine learning to predict which files you're most likely to want to work on next and presents those as a set of cards you can choose from. Only files that you have access to in OneDrive or SharePoint are suggested, which might prove a little frustrating.

Recommended files in Office has been available on Office.com and Office for Mac and is now coming to Office on Windows.

## Beat the Boredom Challenge

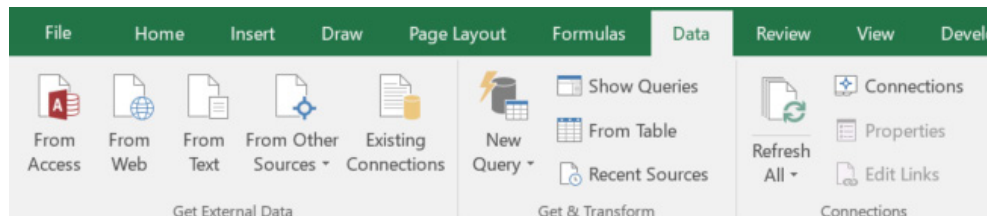
*With many of us currently “working from home” / quarantined, there are only so Zoom / Teams calls and virtual parties you can make before you reach your (data) limit. Perhaps they should measure data allowance in blood pressure millimetres of mercury (mmHg). To try and keep our*

*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...*

Let's talk about refreshing Power Query queries. In VBA, you can trigger a refresh of your data connections, including your Power Query queries, using the following code:

```
Sub RefreshAll()  
'Refresh the queries  
ActiveWorkbook.RefreshAll  
End Sub
```

Seems pretty basic, right? This is basically what happens when you click “Refresh All” in the Data tab in Excel.

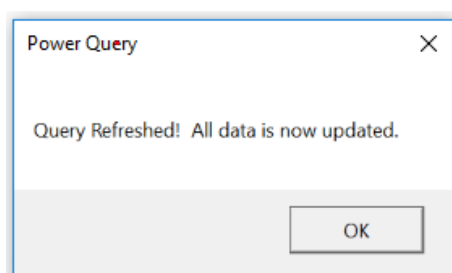


Now, in the bottom left hand side of Excel, when queries are refreshing, you would see a new message come up. This lets you know that queries are refreshing, and allows you to perform other actions in Excel while you're waiting for it to work.



This is actually quite a vast improvement from the old days – you'll see many blogs online complaining that you can accidentally hit 'Esc' and cancel the query refresh – this isn't the case anymore! If you want to cancel your refresh, you can click on the message accordingly.

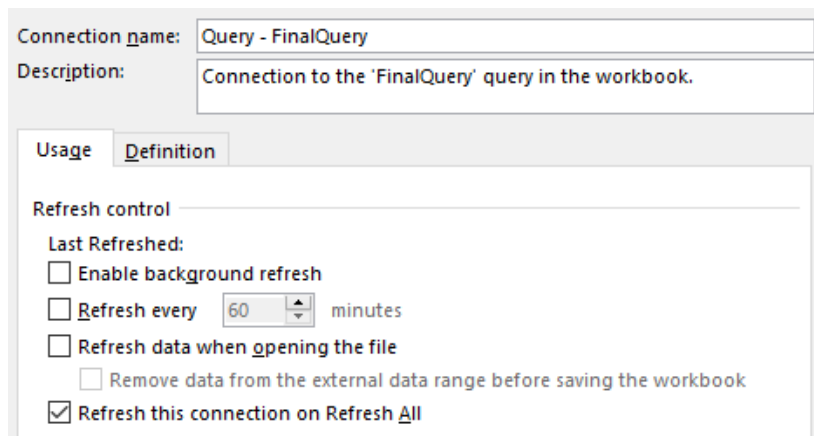
There are two main issues with this. The first is that Power Query (or Get and Transform) doesn't tell you when your queries have finished refreshing. Ever watched a pot of water come to boil? The more you stare at it, the longer it feels like it takes! It'd be nice to have a message box pop up to let you know that your queries are done.



Of course, this leads us to problem two – Power Query doesn't communicate back to VBA. So if I were to update the previous query as follows:

```
Sub RefreshAll()  
'Refresh the queries  
ActiveWorkbook.RefreshAll  
MsgBox "Query Refreshed! All data is now updated."  
End Sub
```

This will trigger the message box to appear after the queries start to refresh, not once they're finished. This can be solved by changing the queries to disable background refresh, but that's not always a practical solution, especially when you're working with large amounts of data, because Excel will be unusable while your VBA script is still running.

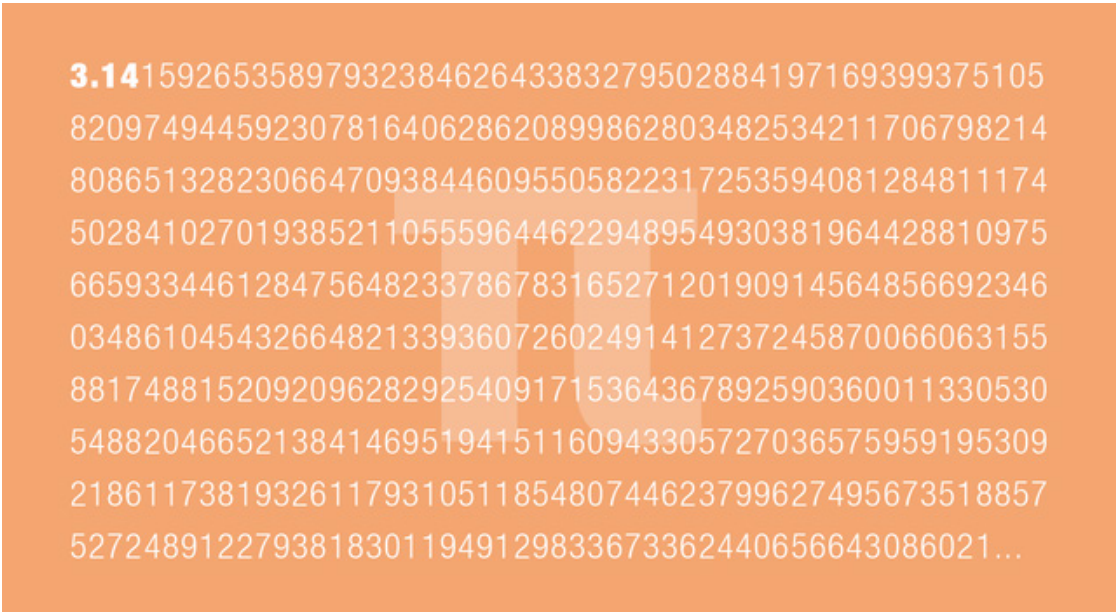


This brings us to our finale: can you create a file that will refresh your Power Query queries, and pop up a message box letting us know when it's finished, without disabling background refresh?

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

## Charts and Dashboards

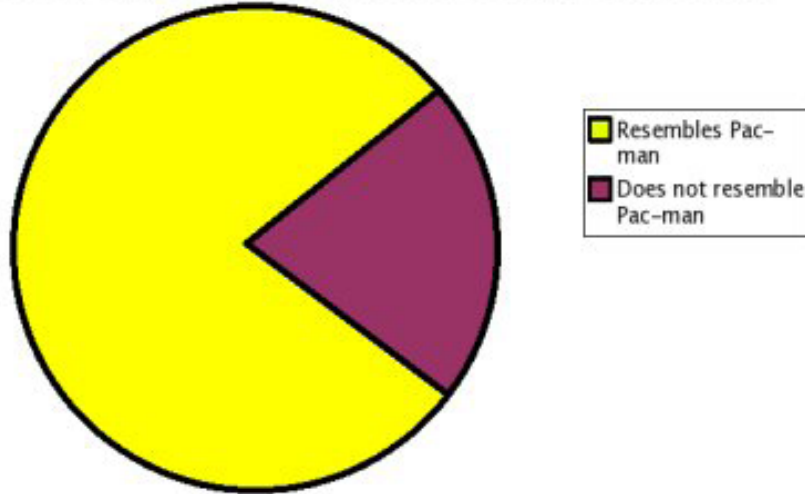
*It's time to chart our progress with an introductory series into the world of creating charts and dashboards in Excel. This month, we look at Pie charts.*



*Not that sort of π...*

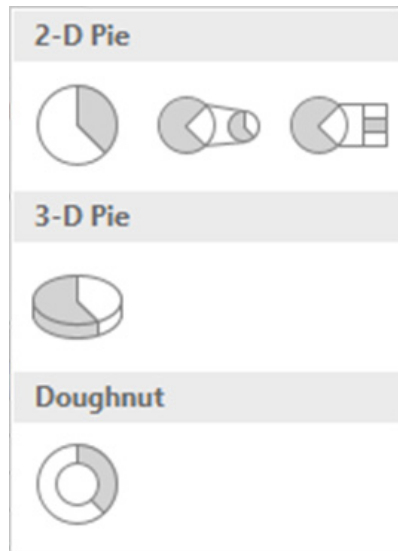
My favourite Pie chart:

## Percentage of Chart Which Resembles Pac-man



As one of the few charts that does not have an axis, a Pie chart is useful for showing the proportion of a group of numbers to their total. And that's about it. Excel also provides the standard 2-D and 3-D Pie charts,

whereby each value in the series is allocated a segment of the circle based upon its ratio of the total.



Note that the two 2-D multiple Pie charts are referred to as 'Pie of Pie' and 'Bar of Pie', and these allow for some of the values in the selected series to be shown as either another Pie Chart or a series of blocks which in total represent one segment of the main Pie Chart. The final option is a Doughnut, which is basically a Pie Chart with a hole in the centre, whereas a Pie chart is a Pie chart with a whole in its centre...

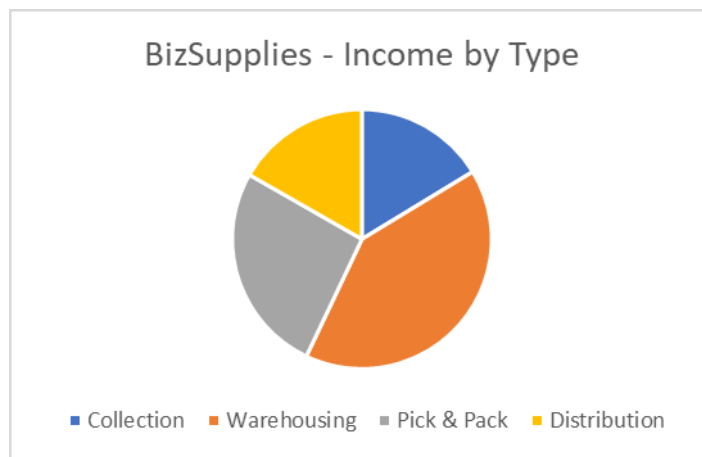
However, there is one key difference between the four Pie charts and the one Doughnut chart: the Pie charts may only graph one series, whereas the Doughnut Chart allows for multiple series to be mapped.

Let's start by taking the 2017/18 income table again:

Income for 2017/18 by Type and Customer					
Income Type	BizSupplies	Dynamic Attire	Harmonic Sonics	Plumb'n' Stuff	Other Clients
Collection	\$ 51,273	\$ 53,709	\$ 40,677	\$ 60,250	\$ 36,616
Warehousing	\$ 128,194	\$ 81,318	\$ 66,315	\$ 91,765	\$ 67,401
Pick & Pack	\$ 82,399	\$ 71,216	\$ 98,094	\$ 112,581	\$ 49,320
Distribution	\$ 52,637	\$ 68,502	\$ 84,749	\$ 57,393	\$ 38,515

I want a Pie chart that shows the distribution of total income by customer and a separate Pie chart of income type by activity. To start, I highlight the table excluding the heading on the top line, then go to the 'Insert'

tab on the Ribbon, click on the 'Pie Chart' icon and choose the top left Pie Chart variation. Here is my initial pie:



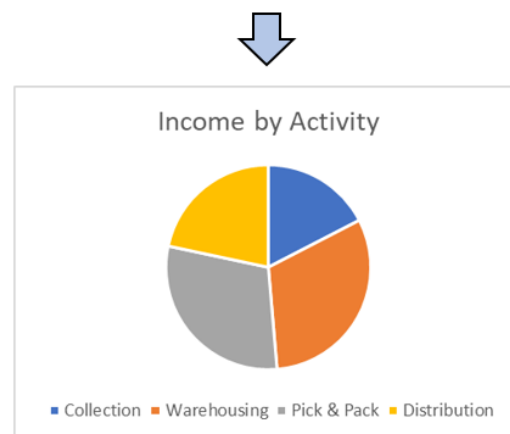
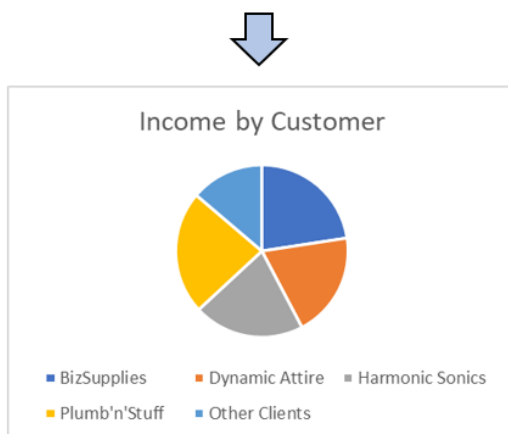
My chart only shows the income type breakdown for BizSupplies because a Pie Chart can only map one series, but the table I highlight could be interpreted as either four series going across the table or five series vertically. Therefore, Excel defaults to selecting the first vertical column as the series for the Pie Chart. Given that I want the data to

be breakdowns of total income, I need to develop data labels to reflect what I want in the charts.

Below is an example of what the data tables would need to look like and their respective Pie Charts:

Income for 2017/18 by Customer	
Customer	Income by Customer
BizSupplies	\$ 314,503
Dynamic Attire	\$ 274,745
Harmonic Sonics	\$ 289,835
Plumb'n'Stuff	\$ 321,989
Other Clients	\$ 191,852

Income for 2017/18 by Type	
Income Type	Income by Activity
Collection	\$ 242,525
Warehousing	\$ 434,993
Pick & Pack	\$ 413,610
Distribution	\$ 301,796

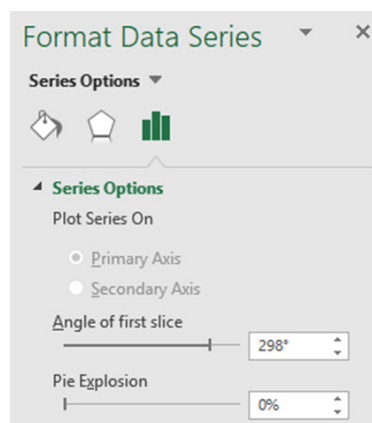


When I take a closer look at the 'Income by Customer' pie, I see little information except that 'Other Clients' and 'Plumb 'n' Stuff' appear to be the largest wedges. Meanwhile, the 'Income by Activity' chart provides more information about the key activities bringing in the majority of income.

However, I would like to have Warehousing as the first wedge. One solution is to alter the data table and put the Warehousing line first, but this would rearrange the series itself, affecting the Legend. A better option may be to simply *rotate* the Pie Chart.

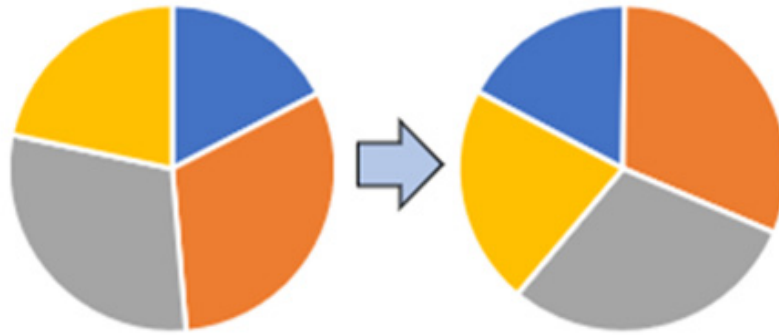
Note that a Pie chart starts plotting the slices from the upper most point, so it draws a line from the centre to the top and then plots each segment going clockwise. Therefore, Collection is first, then Warehousing, etc.

To do this, I click on the Pie Chart so the segments are highlighted, then I right click and choose 'Format Data Series'. Under 'Series Options', I set the 'Angle of the first slice':



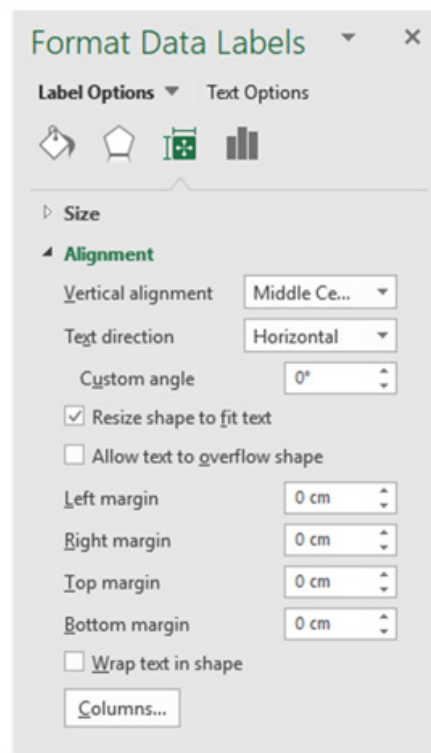
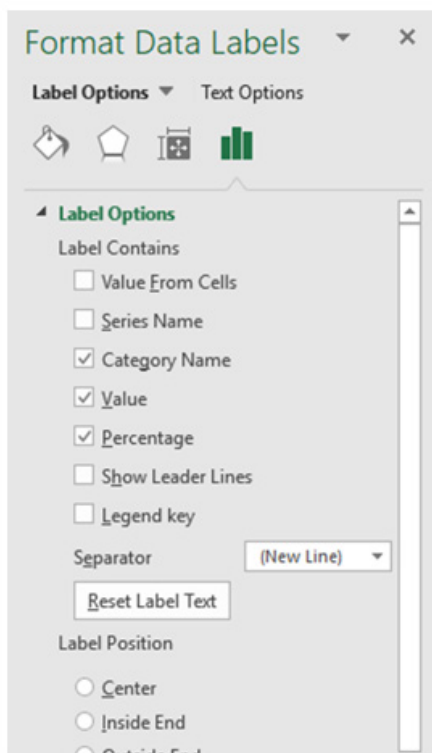


The Pie chart will reflect the change:

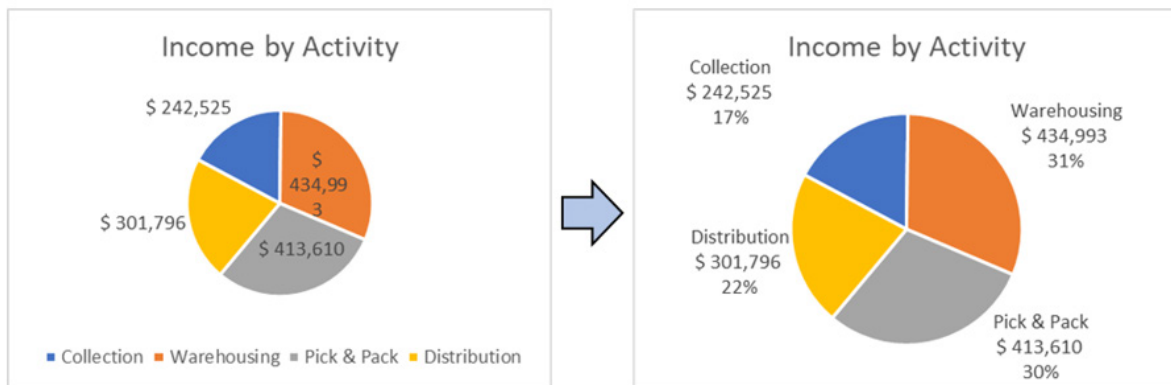


Next, it would be nice to display data information on the Pie chart against each segment, which helps communicate statistics to the user. I right-click on the Pie Chart and choose 'Add Data Labels'. Excel will automatically place the values from the source data next to each segment

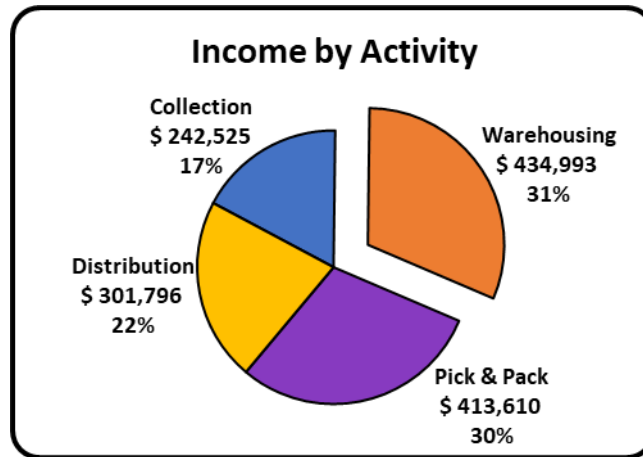
on the chart, but there are a range of choices I can use for data labels. To see them, I right-click on the data labels just added, and choose 'Format Data Labels', then apply 'Category Name', Value and Percentage. Then, I set the following text options under 'Size & Properties':



Since I have the categories in the data labels, I no longer need the Legend, so I click on the Legend and delete it.



It can be clearly seen from the chart that Warehousing is the activity that generate the majority of income, so, I want to make it stand out. I double click on the Warehousing wedge, hold and pull it outwards then release the mouse. I do some further formatting and here is the Pie I've made:



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 review `DataBodyRange`.

Last month, we modified the contents of a table, but now we will learn how use the `DataBodyRange` property to manipulate table ranges.

the main body of data, i.e. it includes all of the rows of data, but does not refer to the headers or any subtotals. For example, say we have the following table, with the table name of `StudentScores`:

The `DataBodyRange` property refers to the part of a Table that contains

ID	Name	Score
1	Julie Wight	43
2	Kimberly Haynes	74
3	Crystal Guerrero	73
4	Eugene Woods	55
5	Larry Fuller	78
6	Charlene Fuller	32
7	Joni Dash	31
<b>Average</b>		<b>55</b>

Let's define our table using the following lines of code in VBA:

```
Dim MyTable As ListObject
Set MyTable = Range("StudentScores").ListObject
```

If we want to copy this entire table and paste it somewhere else, we can use this code:

```
MyTable.DataBodyRange.Copy
Range("F3").PasteSpecial xlPasteAll
```

Once we run the macro, our sheet should look something like this:

	A	B	C	D	E	F	G	H
1								
2		ID	Name	Score				
3		1	Julie Wight	43		1	Julie Wight	43
4		2	Kimberly Haynes	74		2	Kimberly Haynes	74
5		3	Crystal Guerrero	73		3	Crystal Guerrero	73
6		4	Eugene Woods	55		4	Eugene Woods	55
7		5	Larry Fuller	78		5	Larry Fuller	78
8		6	Charlene Fuller	32		6	Charlene Fuller	32
9		7	Joni Dash	31		7	Joni Dash	31
10		<b>Average</b>		<b>55</b>				

We can also use the **DataBodyRange** property to resize tables. Using the following code, we can keep the last three [3] rows of the table and delete the rest:

```
MyTable.DataBodyRange.Resize(MyTable.DataBodyRange.Rows.Count - 3, _  
MyTable.DataBodyRange.Columns.Count).Rows.Delete
```

To explain the syntax:

- `Resize(MyTable.DataBodyRange.Rows.Count - 3)`: this counts the total number of rows in the table and subtracts three [3], which means that the last three rows will not be deleted
- The underscore next to "-3" simply tells VBA that we're not done with the line of code yet, allowing us to organise code more efficiently.

After running the code above, we should end up with a table looking like this:

ID	Name	Score
5	Larry Fuller	78
6	Charlene Fuller	32
7	Joni Dash	31
<b>Average</b>		<b>47</b>

That's it for this VBA article. Join us next time for more on the DataBodyRange property!

More next time.

## Power Pivot Principles

We continue our series on the Excel COM add-in, Power Pivot. This month, we discuss why it is important to have contiguous dates in a calendar table.

Just as a reminder, a **contiguous** date table is a table that stores all of the dates with no gaps (*i.e.* no missing dates). In addition to being contiguous the table's dates also need to be in strict ascending order and contain no duplicates. That means 1, 2, 3, 4, 5 is fine, but 1, 2, 3, 5 and 1, 2, 3, 3, 5 are not.

With that borne in mind, let's go back to our past example using the

**TOTALYTD** function. This time, we will create a measure with the **TOTALYTD** function where the dates expression is not linked to a contiguous date table; instead, we'll link it to the **OrderDate** column in the **Sales** table (*say*).

Therefore, that the measure is easily recognisable; we will call this measure '**Total Sales YTD 3**':

The screenshot shows the 'Measure' dialog box in Excel. The 'Table name' is 'Sales', the 'Measure name' is 'Total Sales YTD 3', and the 'Description' is empty. The 'Formula' field contains '=TOTALYTD([Sales], 'Sales'[OrderDate])'. The 'Formatting Options' section shows the 'Category' set to 'Currency', the 'Symbol' set to '\$', and 'Decimal places' set to 2. The 'Use 1000 separator' checkbox is checked. The 'OK' and 'Cancel' buttons are at the bottom right.

The resulting PivotTable will look like this:

	A	B	C	D	E
2					
3		<b>Year</b>	<b>Month</b>	<b>Sales</b>	<b>Total Sales YTD 3</b>
4		⊕ 2014		\$29,502.45	\$29,502.45
5		⊖ 2015	1	\$5,231.30	\$5,231.30
6			2	\$5,097.90	\$5,097.90
7			3	\$5,667.55	\$5,667.55
8			4	\$6,063.90	\$6,063.90
9			5	\$6,029.40	\$6,029.40
10			6	\$5,752.15	\$5,752.15
11			7	\$7,669.80	\$7,669.80
12			8	\$9,016.25	\$9,016.25
13			9	\$5,861.15	\$5,861.15
14			10	\$7,105.85	\$7,105.85
15			11	\$7,008.80	\$7,008.80
16			12	\$10,197.60	\$10,197.60
17		⊕ 2016		\$734,038.55	\$734,038.55
18		⊕ 2017		\$976,355.90	\$976,355.90
19		<b>Grand Total</b>		<b>\$1,820,598.55</b>	<b>\$976,355.90</b>

Do you see that the 'Total Sales YTD 3' measure returns with the actual sales for each of the months and not the year to date sales? This is because the dates in the **OrderDate** column are not contiguous (there are dates

with gaps in them, for example), thus causing the functions in Power Pivot to return with unexpected results. Using a contiguous date table instead will allow Power Pivot to correctly calculate the YTD figures:

<b>Year</b>	<b>Month</b>	<b>Sales</b>	<b>Total Sales YTD</b>
⊕ 2014		\$29,502.45	\$29,502.45
⊖ 2015	1	\$5,231.30	\$5,231.30
	2	\$5,097.90	\$10,329.20
	3	\$5,667.55	\$15,996.75
	4	\$6,063.90	\$22,060.65
	5	\$6,029.40	\$28,090.05
	6	\$5,752.15	\$33,842.20
	7	\$7,669.80	\$41,512.00
	8	\$9,016.25	\$50,528.25
	9	\$5,861.15	\$56,389.40
	10	\$7,105.85	\$63,495.25
	11	\$7,008.80	\$70,504.05
	12	\$10,197.60	\$80,701.65
⊕ 2016		\$734,038.55	\$734,038.55
⊕ 2017		\$976,355.90	\$976,355.90
<b>Grand Total</b>		<b>\$1,820,598.55</b>	

More *Power Pivot Principles* next month.

## Power Query Pointers

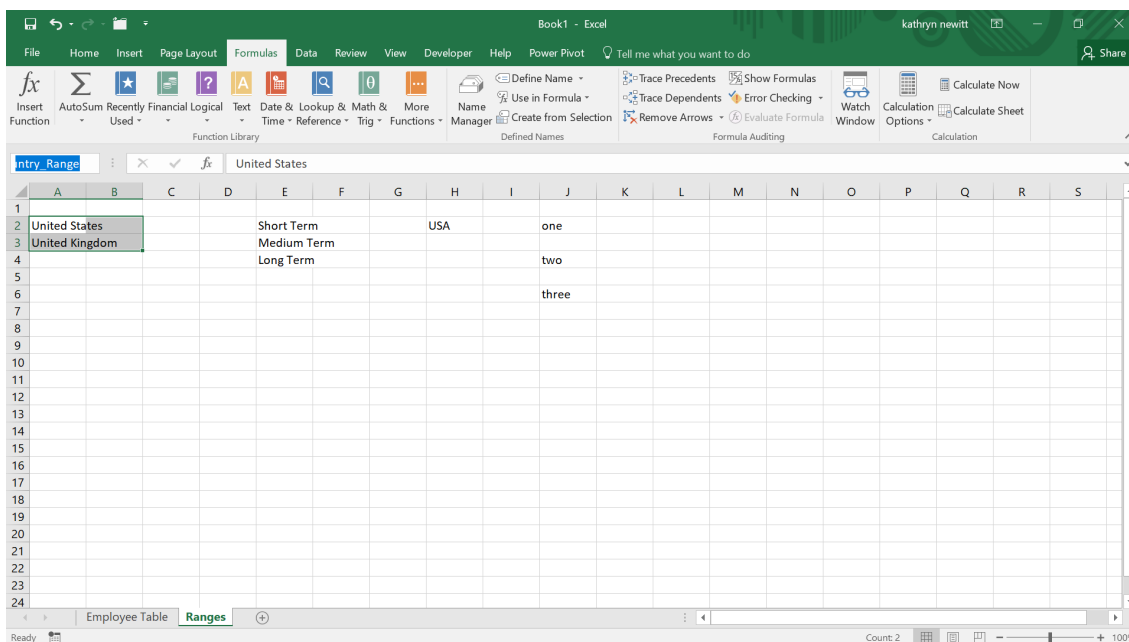
Each month we'll reproduce one of our articles on Power Query (Excel 2010 and 2013) / Get & Transform (Office 365, Excel 2016 and 2019) from [www.sumproduct.com/blog](http://www.sumproduct.com/blog). If you wish to read more in the meantime, simply check out our Blog section each Wednesday. This month, we take another look at what you may do with Excel.CurrentWorkbook().

Previously, we have seen how to reference a single cell in a worksheet. I'd like to return to the **M** function **Excel.CurrentWorkbook** to show how to reference more than one cell. To begin with, the description of **Excel.CurrentWorkbook** in the Microsoft help pages doesn't give much away:

### **Excel.CurrentWorkbook() as table**

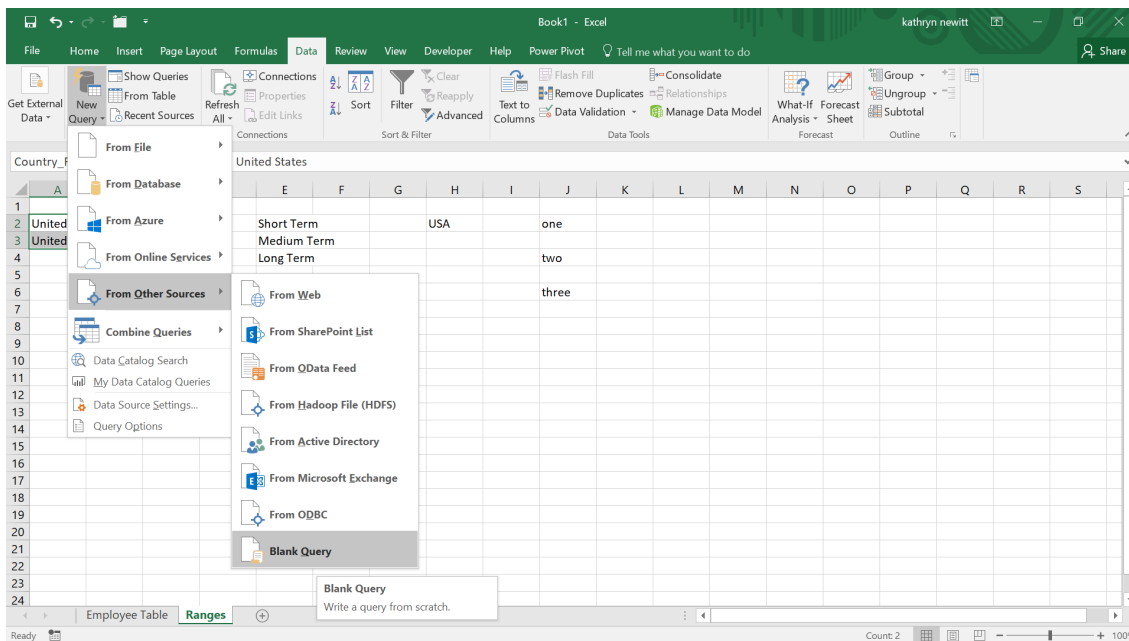
This returns the tables in the current Excel Workbook.

What this doesn't reveal, is that it is not just the tables that are returned, but also any named ranges. Let's consider a workbook containing some fictional employee data, which has a table on one sheet and named ranges on another sheet, as shown below:

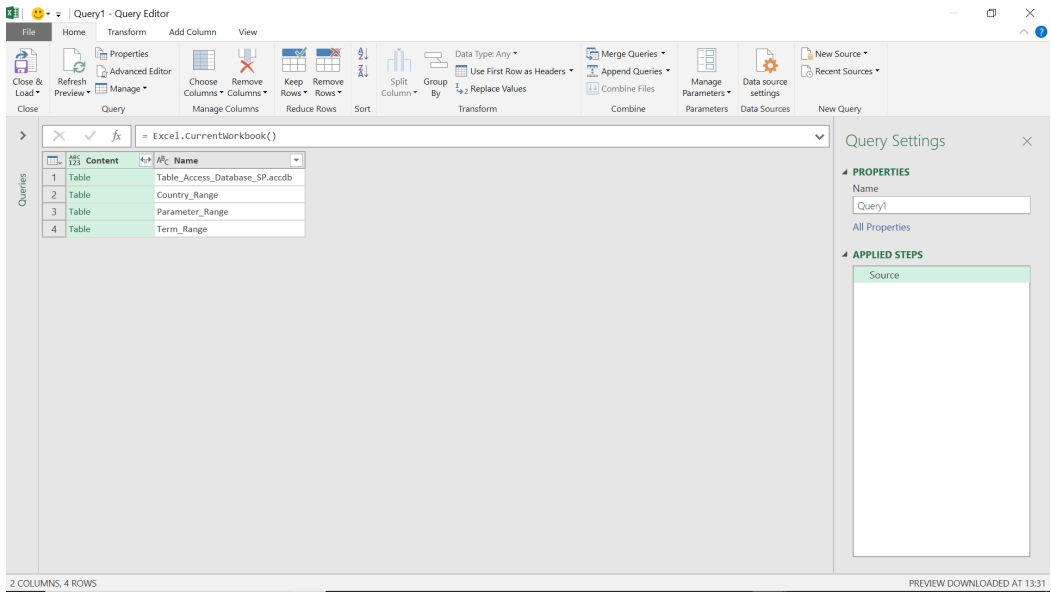


We might wish to pull data from these ranges into a query on employee data, to use for parameter selection, for example. If we create a Blank Query, we may use the **Excel.CurrentWorkbook()** function to show me

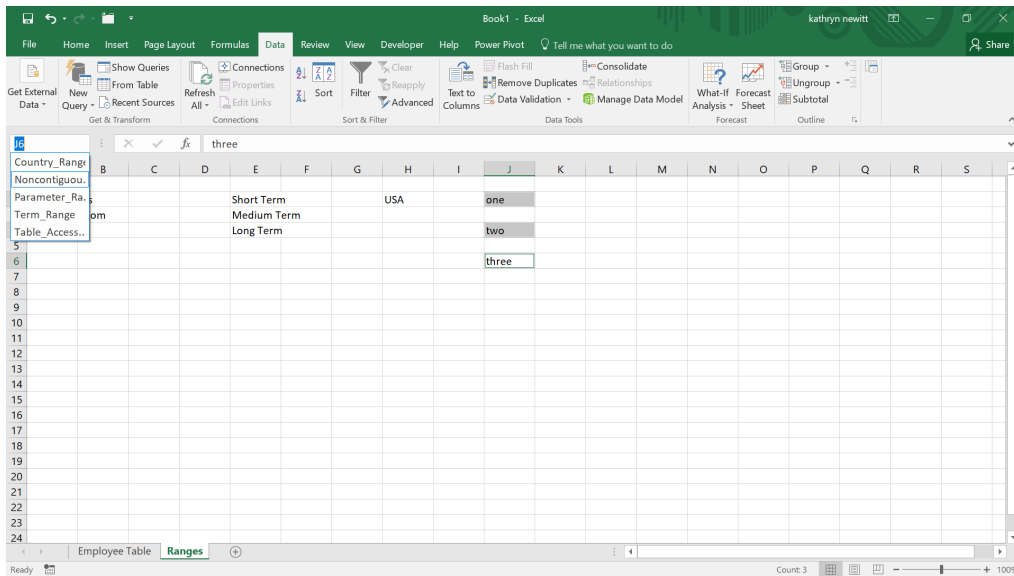
what tables and ranges are available. To create a Blank Query, in the Data tab, choose the 'New Query' option and select 'Blank Query' from the 'From Other Sources' section, viz.



Now we may use the function **Excel.CurrentWorkbook()**.

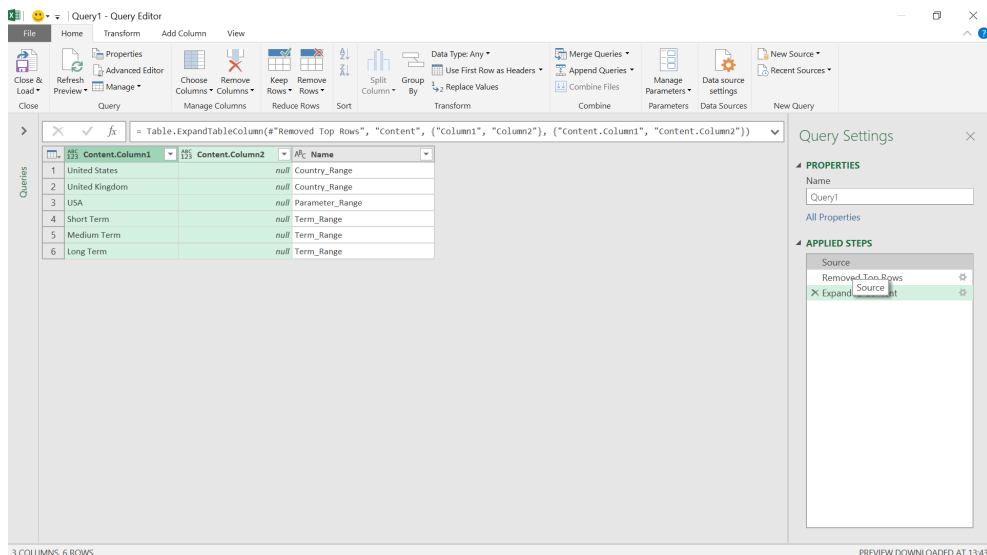


At the top, we have our **Employees** table, named according to the Access database it has been extracted from. Below that there are three ranges. There is a range missing from our workbook, and it is missing because of a particular property.

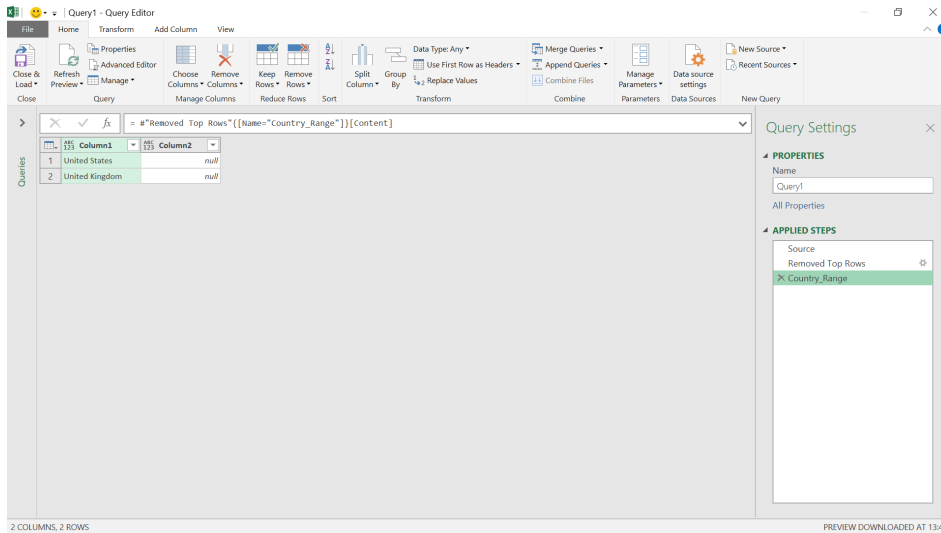


We created a non-contiguous range in our workbook (where the cells are not consecutive) and this hasn't been picked up by Power Query. The previous version of Power Query dealt inconsistently with these types of ranges, which is probably why they are no longer accessed by the function `Excel.CurrentWorkbook()`.

Back in our query, we will remove the row relating to the **'Employee Table'** and use the split arrow icon to drill down to the content of the named ranges:

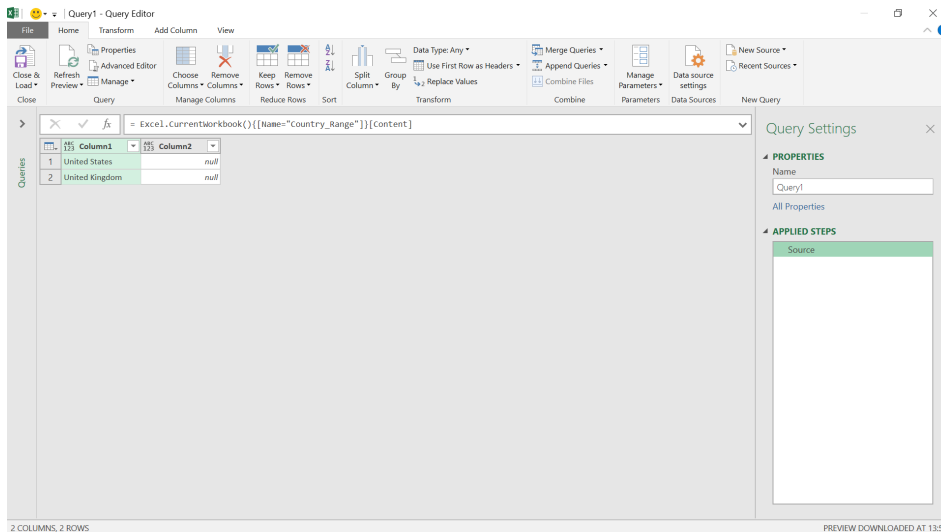


This shows that all our data is accessible. If we want to select data from a particular named range, then instead of using the expand icon, we may click on the word 'Table' next to the range required.



It should be noted that when you try this, Power Query may promote the top row to a heading – if this happens, just delete that step.

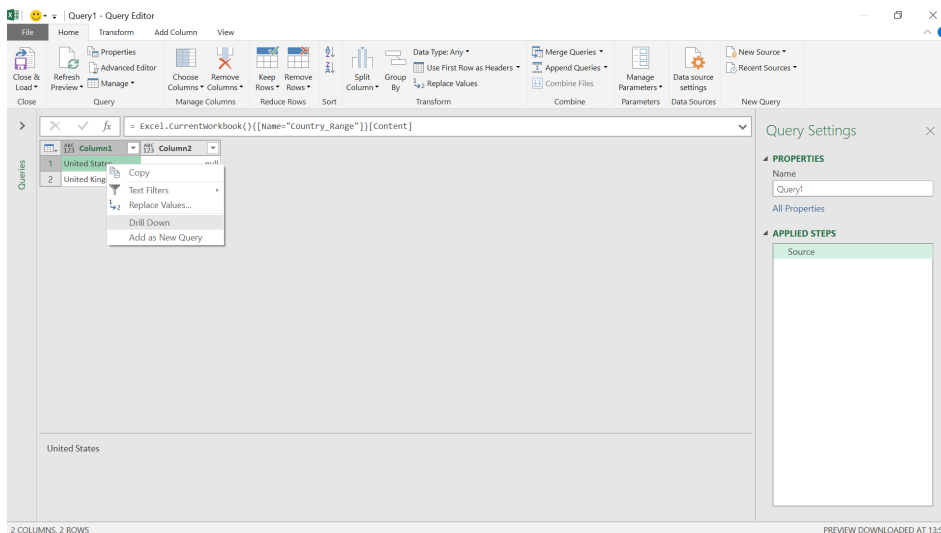
We may also get to this data at the beginning of the query creation if we know the name of the range required.



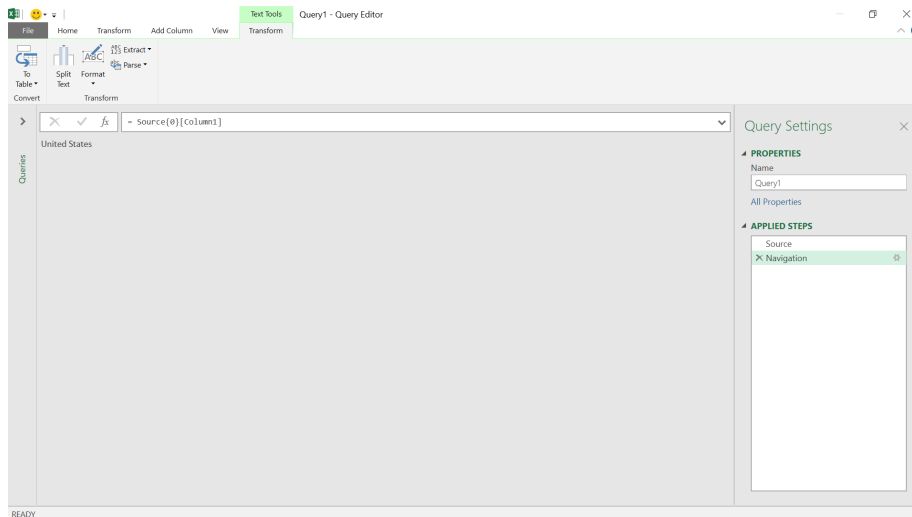
The M language to access my 'Country\_Range' is

**= Excel.CurrentWorkbook() { [Name="Country\_Range"] } [Content]**

To access a single cell, e.g. 'United States', we may right click on the value and use the 'Drill Down' option:



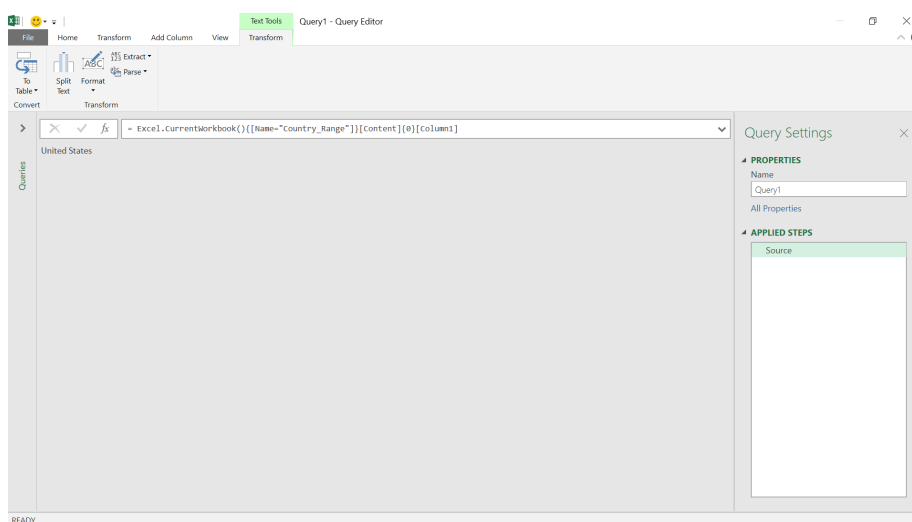
This extracts a single value:



This is the functionality that was used previously, but amended to look at **Country\_Range**:

**Excel.CurrentWorkbook() { [Name="Country\_Range" ] [Content] {0} [Column1]**

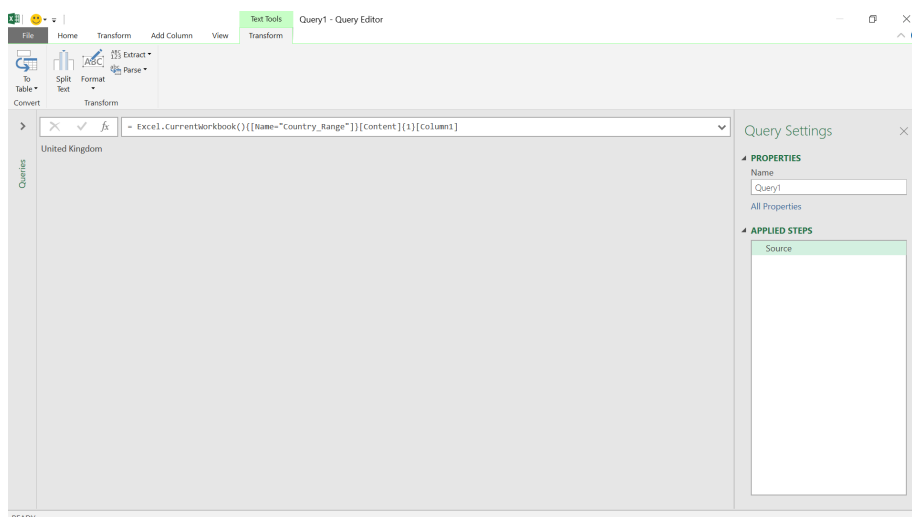
If you enter this at the beginning of the query, you may go straight to the value in that cell:



Note that in the function,

**Excel.CurrentWorkbook() { [Name="Country\_Range" ] [Content] {0} [Column1]**

the 0 highlighted in **red** text indicates the value on the first row; if we wanted the second value, we would enter 1 instead (Power Query likes to index many lists starting at zero, rather than one).



More next month!



# Power BI Updates

This month's updates see the release of additional functionality and performance improvements for DirectQuery, optimisation for the SWITCH function, new Bitwise DAX functions, and the General Availability of the Premium Gen2 platform for Premium capacities.

The full list reads as follows:

## Reporting

- Heat map layer (Azure Maps Visual) in Preview
- DirectQuery for Power BI datasets and Azure Analysis Services in Preview:
  - More flexibility when selecting tables
  - Performance improvements
- Tenant-level feature switch for Bing Maps

## Modelling

- SWITCH performance optimisation
- Bitwise functions in DAX

## Data Connectivity and Preparation

- Amazon Redshift (connector update)
- Dataverse (connector update)
- Google Analytics (connector update)
- Azure Cosmos DB V2 (new connector) in Preview
- Azure Databricks (connector update)
- Amazon Athena now Generally Available
- SIS-CC-SDMX (new connector) in Preview
- SumTotal (connector update) now Generally Available
- Usercube (new connector) in Preview

## Service

- Administration and governance:
  - Premium Gen2 now Generally Available
  - New API for determining who has access to what Power BI assets in Preview

- Deployment pipeline APIs now Generally Available
- Upcoming automatic installation of the Power BI app for Microsoft Teams

## Embedded Analytics

- Export Power BI report to file API: Preview update
- Object-Level Security (OLS) now Generally Available in Power BI Embedded
- Adjust zoom level for embedded Power BI reports
- Updated Multi-Geo support for Power BI Embedded
- Power BI Embedded Playground update: use your own embed token
- Power BI Angular component

## Developers

- Add modern ToolTip to your custom visual
- Modal Dialog API update

## Visualisations

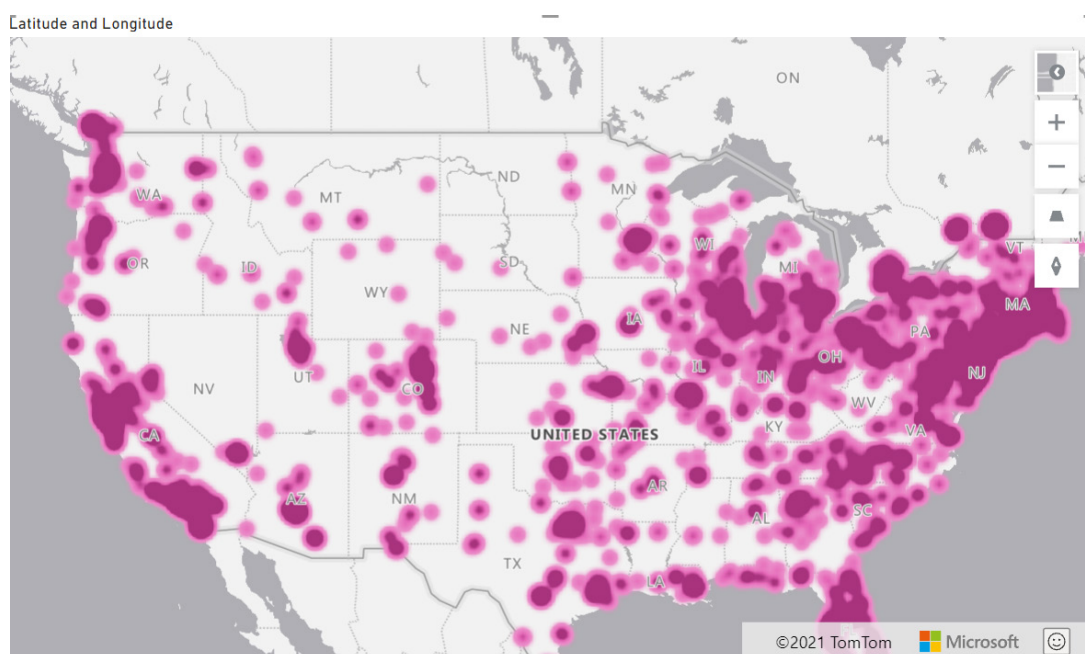
- Lollipop Bar Chart by Nova Silva
- Drill Down TimeSeries PRO visual by ZoomCharts
- Inforiver by Lumel
- Hierarchical bar chart by Excelnaccess.com
- New visuals in AppSource

## Other

- ODBC support for paginated reports now Generally Available.

Let's go through each in turn.

## Heat map layer (Azure Maps Visual) in Preview



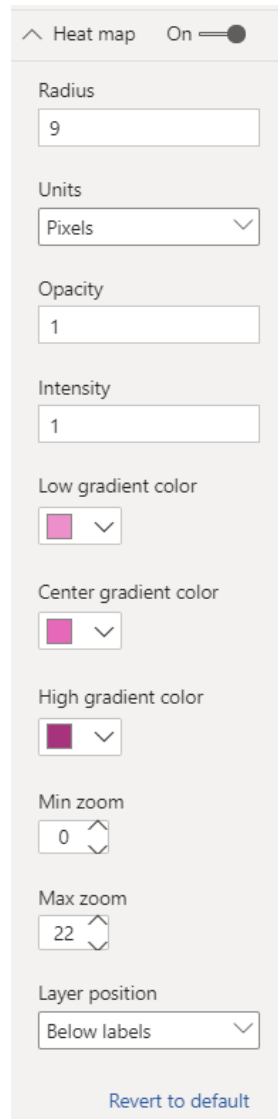
This update sees the introduction of the heat map layer to the Azure Maps Visual.

The heat map displays colour patterns on the map to represent the density of data points, using a range of colours, showing the data "hot spots" on a map.

Azure Maps is in Preview, so to switch it on, go to **File -> Options and settings -> Options -> Preview features** and then toggle the Azure Maps Visual to On.

The Format pane for Heat map lets users customise and design the heat map visualisations. The Format pane options allow you to:

- configure the radius of each data point using pixels or meters as units
- customise the opacity and intensity of the heat map layer
- specify whether the value in the size field should be used as the weight of each data point
- select different colours from colour pickers drop-down
- set minimum and maximum zoom levels for the heat map layer to be displayed
- decide the heat map layer position among different layers, e.g. 3D bar chart layer and bubble layer.



A heat map is useful when users want to visualise comparative data, such as:

- comparing customer satisfaction rates or shop performance among regions or countries
- measuring the frequency which customers visit shopping malls in different locations.
- visualising statistical / geographical data sets.

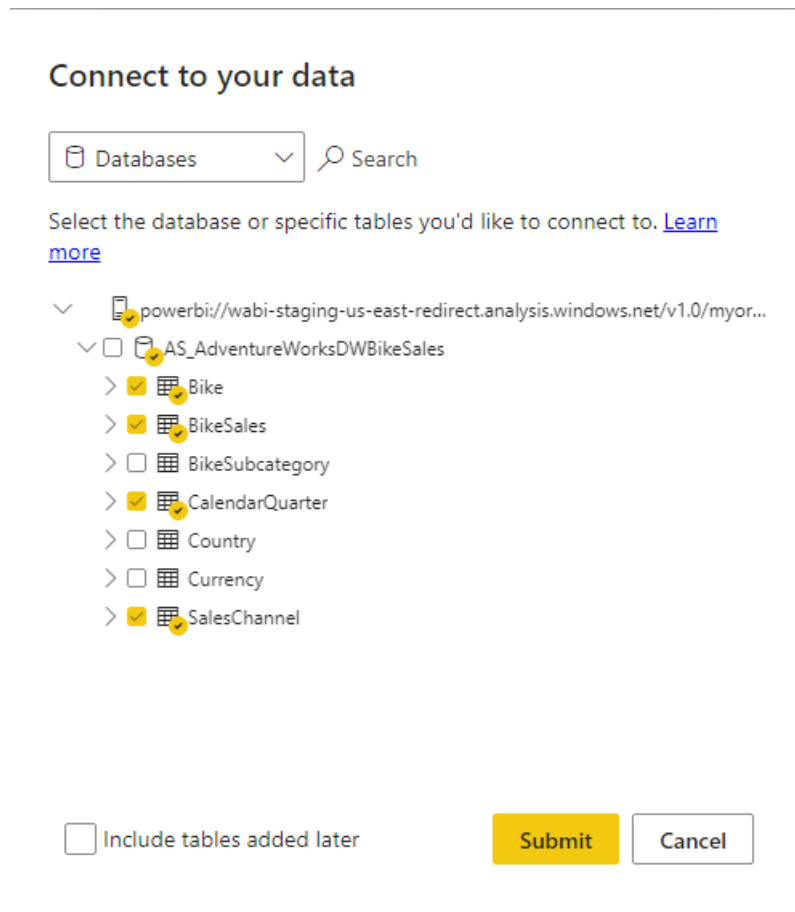
### ***DirectQuery for Power BI datasets and Azure Analysis Services in Preview***

#### **MORE FLEXIBILITY WHEN SELECTING TABLES**

Previously, when you made a DirectQuery connection to a dataset or Azure Analysis Services model, your model contained all tables. This release provides you with greater control, with two new features:

1. You may now remove a table from your model using the Field list. This way, you can keep your model clean, even if there are more tables in the dataset or model you are connected to. (however, it should be noted that if you connect to a perspective the ability to remove tables from your model is not available to you)

- When connecting to a dataset or model you may now specify exactly which tables to load in the first place. There is no longer a need to load everything if you are only looking for a specific subset of tables.



Note that you can also choose to automatically add any table that might get added to the dataset or model after you make the connection to your model. You should be aware that when you connect to a perspective, your model will contain all tables in the dataset or model and any table not included in the perspective will be hidden. Moreover, any tables that might get added to the perspective will be added automatically.

This dialog will only show if you add a DirectQuery connection to a Power BI dataset or Azure Analysis Services model to an existing model. You can also open this dialog by changing the DirectQuery connection to the Power BI dataset or Azure Analysis Services model in the Data source settings after you created it.

## PERFORMANCE IMPROVEMENTS

Parallel execution of model queries and smart caching has improved performance across the board.

### *Tenant-level feature switch for Bing Maps*

November will see the introduction of a tenant-level feature switch to enable or disable the Bing Maps visual. You will find this new option in the Admin Portal. Moving forward, Bing Maps will require explicit opt-in for new tenants, meaning it will be disabled by default. Existing tenants

should not be affected, but you should note that the option will now exist to disable the visual. Loading Bing Maps visuals in a context where they have been disabled will result in an error only affecting that visual, which notifies the user to contact their admin to opt in.

### *SWITCH performance optimisation*

Microsoft has optimised the performance of the **SWITCH** function in DAX. You should see the improved performance if the **SWITCH** function has a

large number of values (hundreds or more) or if the value expressions contain measure references.

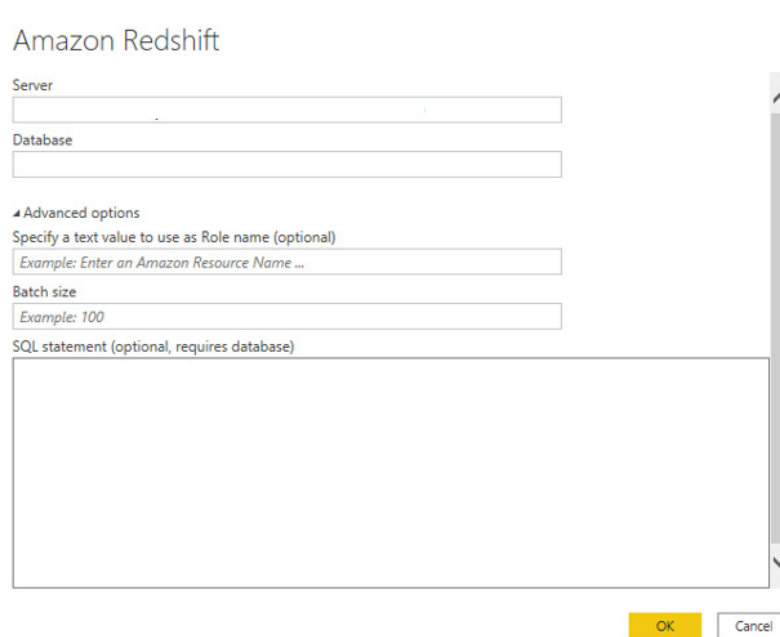
### *Bitwise functions in DAX*

DAX now includes some new Bitwise functions. You may now perform common bitwise operations, including “shifting” in DAX. The following functions have been added:

- BITLSHIFT** and **BITRSHIFT**, which allow you to shift a number left or right by a specified number of bits respectively
- BITAND**, **BITOR** and **BITXOR** return bitwise versions of **AND**, **OR** and **XOR** respectively of two numbers respectively.

### **Amazon Redshift (connector update)**

There is now support for native database queries for the Amazon Redshift connector. Similar to SQL and Snowflake connectors, this will allow you to input an Amazon Redshift native query and build reports on top of it.



The screenshot shows a configuration window titled "Amazon Redshift". It contains several input fields: "Server", "Database", "Advanced options" (with a sub-label "Specify a text value to use as Role name (optional)" and an example "Example: Enter an Amazon Resource Name ..."), "Batch size" (with an example "Example: 100"), and a large text area for "SQL statement (optional, requires database)". At the bottom right, there are "OK" and "Cancel" buttons.

### **Dataverse (connector update)**

Microsoft has updated the Dataverse connector to support environment discovery. Users will now be able to view a list of environments that they have access to, instead of being required to specify an environment URL.

### **Google Analytics (connector update)**

The Google Analytics Connector has been updated to use V4 APIs. Additional features in this update include better API reliability and exposure of additional dimensions and measures not available in V3.

### **Azure Cosmos DB V2 (new connector) in Preview**

To query data stored in the transactional store of Azure Cosmos DB containers, Power BI had previously released a native Power BI connector (henceforth referred to as Azure Cosmos DB V1 connector) that was Generally Available and supported only Import mode. This update releases a new native Power BI connector (henceforth referred

to as Azure Cosmos DB V2 connector) that will support querying the transactional store in both DirectQuery and Import modes. In addition to supporting DirectQuery mode, the V2 connector includes performance optimisations related to query pushdown and data serialisation, whatever that is.

### **Azure Databricks (connector update)**

The Azure Databricks connector has been updated. This has added support for navigation through catalogue hierarchy in workspaces with Unity Catalog support. It has enabled 'Fast Evaluation' by default,

providing faster processing of large imports, and enabling direct SQL passthrough, to provide lower latency for Databricks SQL and Databricks Runtime 8.3 and later.

### **Amazon Athena now Generally Available**

This connector is now Generally Available.

### **SIS-CC-SDMX (new connector) in Preview**

The SIS-CC SDMX connector allows to easily query any public SDMX REST web service endpoint that supports the SDMX-CSV output format to analyse, visualise and share data from Power BI. Statistical Data and Metadata eXchange (SDMX) is an ISO international open standard adopted and used by statistical organisations (national, sub-national,

regional and international) for the production and dissemination of official statistics.

The SIS-CC SDMX connector is an open-source project provided by the Statistical Information System Collaboration Community (SIS-CC).

## ***SumTotal (connector update) now Generally Available***

The SumTotal connector has been updated and is now Generally Available. The updates include:

- removing hardcoded list of OData entities; the connector now pulls all OData APIs dynamically
- adding logic to allow bypassing of SSO login via query string parameter.

## ***Usercube (new connector) in Preview***

Usercube is the first no-code full-featured solution for Identity Governance and Administration (IGA), configurable to your organisation, process and policies.

All users, including the internal workforce, partners, RPA bots and IoT devices, automatically receive the necessary access permissions, enforcing

your security policies and the segregation of duties (SoD) based upon your unique role model. With Usercube's Power BI integration, business reports benefit from data such as users' identities, entitlement details (accounts and fined-grained permissions) and approval workflow tracing.

## ***Administration and governance***

### **PREMIUM GEN2 NOW GENERALLY AVAILABLE**

The new second generation platform for Premium Capacities (known as Premium Gen2) is now Generally Available. This new platform provides improved resiliency and performance stability, greater scale of usage with fewer restrictions on memory utilization and concurrent refreshes, modernised monitoring of capacity metrics and optional sizing flexibility with Autoscale capacity sizing. It can increase your capacity

size automatically to address unplanned spikes in demand for capacity processing power.

All new premium capacities will be created as Gen2 capacities, and all existing capacities are encouraged to switch to Gen2. In the next few weeks, Microsoft will automatically port legacy first generation capacities to Gen2.

### **NEW API FOR DETERMINING WHO HAS ACCESS TO WHAT POWER BI ASSETS IN PREVIEW**

Last month a Preview set of APIs were released to retrieve user permissions. This update sees another API available to help you better manage your Power BI asset inventory. The `GetUserArtifactAccessAsAdmin`

API takes the graph ID of a user and returns lists of assets that the user has access to. The results are paginated by a continuationToken, with each page of results containing one type of asset.

## ***Deployment pipeline APIs now Generally Available***

A while ago, there was a Preview release of the deployment pipeline APIs. The deployment APIs enabled BI developers to manage Power BI releases as part of their DevOps tool of choice.

All of these deployment pipeline APIs are now Generally Available. As part of this update, Microsoft is also releasing a new set of APIs to create and manage pipelines from start to finish so it's faster and easier than ever to migrate and scale with Power BI deployment pipelines.

With the new set of APIs, pro BI developers will be able to:

- automatically add new workspaces to work with pipelines: you can set an automated process where new production workspaces are assigned to a pipeline and all relevant team members get the appropriate permissions
- standardise release processes in the organisation: you can migrate content at scale to work with pipelines, so that your release process is up to the highest standards
- central management of Power BI content in your DevOps tool of choice, without opening Power BI: from pipeline creation to deploying to production, all operations can be performed via automation.

The new APIs will allow you to:

- create / update / delete a deployment pipeline
- assign / un-assign a workspace to a deployment pipeline
- add / remove user from a deployment pipeline.

## ***Upcoming automatic installation of the Power BI app for Microsoft Teams***

Many organisations use modern collaboration in Microsoft Teams to enable faster decision making and action. Now, Power BI is making it easier for entities to roll-out Power BI experience in Teams, so that users may discover and use data where they work.

To enable organisations to more easily roll-out of Power BI in Teams, Microsoft has made a few improvements:

- Power BI will begin automatically installing the Power BI app for Teams for users when they visit the Power BI Service
- Power BI administrators may choose not to auto-install through a new Power BI tenant setting
- The tenant setting has started to roll-out now, giving administrators time to opt-out if desired
- The automatic installation will start to take effect this month, for organisations with the setting enabled.

The 'Install Power BI app for Microsoft Teams automatically tenant' setting is added to the Power BI admin portal. Power BI admins can control the auto-install behaviour. By default, the auto-installation is enabled.

#### Install Power BI app for Microsoft Teams automatically *Enabled for the entire organization*

The Power BI app for Microsoft Teams is installed automatically for users when they use the Power BI service. The app is installed for users if they have Microsoft Teams and Power BI is allowed in the Teams Admin Portal. When the app is installed, users receive notifications in Microsoft Teams and can more easily discover and collaborate with colleagues. [Learn more.](#)

Enabled

Apply to:

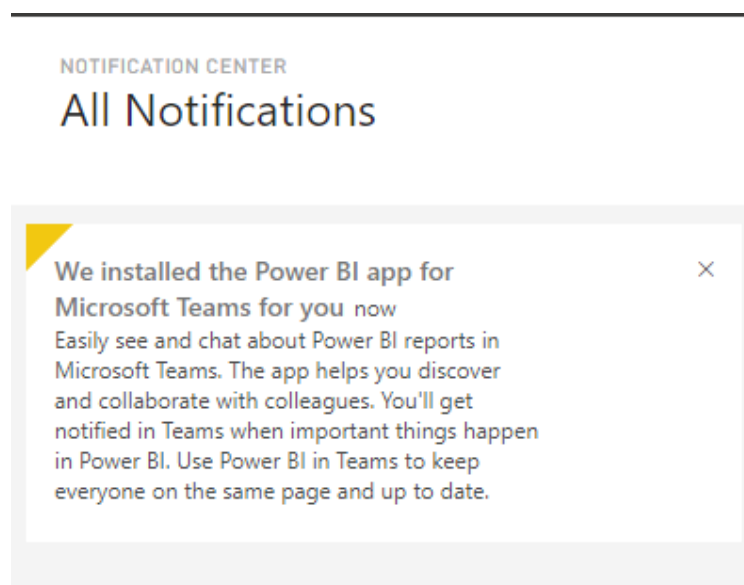
- The entire organization
- Specific security groups
- Except specific security groups

The automatic installation happens for a user under the following conditions:

- the Power BI app for Microsoft Teams is set to allowed in the Microsoft Teams admin portal
- the Power BI tenant setting Install Power BI app for Microsoft Teams automatically is enabled
- the user has a Microsoft Teams license
- the user opens the Power BI service (e.g. [app.powerbi.com](https://app.powerbi.com)) in a web browser.

Initially, auto-install applies to new users the first time they visit the Power BI service in a web browser. In the future, auto-install will occur for all active users of the Power BI service who meet the criteria.

When auto-install occurs, the following notification is shown in the Power BI service notification pane:



#### **Export Power BI report to file API: Preview update**

Export Power BI report to file API (Preview) allows you to export Power BI reports to the following file formats: PDF, PPTX (PowerPoint) and PNG, by using a REST call. Since this API was launched, Power BI has had to impose a few limitations which are now being removed for Premium / Embedded Gen2 capacities. The following modifications apply:

- the number of Power BI exports per hour is not limited to 50 per hour per capacity, but only to 50 report pages per minute per capacity
- there is no maximum concurrent report pages limitation
- an export API operation load is evaluated as a slow-running / background operation as described in Premium Gen2 capacity load evaluation. The workload under which export API operations appears is called 'InteractiveReportExports'.

Microsoft anticipate that this Export Power BI report to file API will become Generally Available by the end of the calendar year.

## Object-Level Security (OLS) now Generally Available in Power BI Embedded

Aligning with Power BI Premium and Pro, Microsoft is moving the object-level security feature from public Preview to General Availability in Power BI Embedded as well. Object-level security (OLS) enables model authors to hide sensitive tables or columns from report viewers. From the standpoint of the user without proper access, the secured table or column does not exist. In addition, object names and metadata are

also secured to prevent a malicious user from discovering that such an object exists. This added layer of security prevents users without the appropriate access levels from discovering business critical or sensitive personal information. Authoring OLS rules in the Power BI dataset can be performed with tools that utilise the XMLA endpoint.

## Adjust zoom level for embedded Power BI reports

You can now set the zoom level of your embedded report programmatically, by either adding a single parameter to the embed settings or by using the setZoom API to adjust the zoom level after your report has loaded. This lets you adjust your report's zoom level to be

different than the default zoom level and to ensure that your report can be fully seen and utilized by all users, regardless of any visual or physical impairments.

To adjust the zoom level from the embed settings, see the following example:

```
let embedConfig = {  
  ...  
  settings: {  
    zoomLevel: 2 // 200% zoom  
  }  
};
```

To get the current zoom level or update it after the report has loaded, use the getZoom and setZoom APIs, as exemplified below:

```
let currentZoom = await report.getZoom();  
await report.setZoom(0.5); // 50% zoom
```

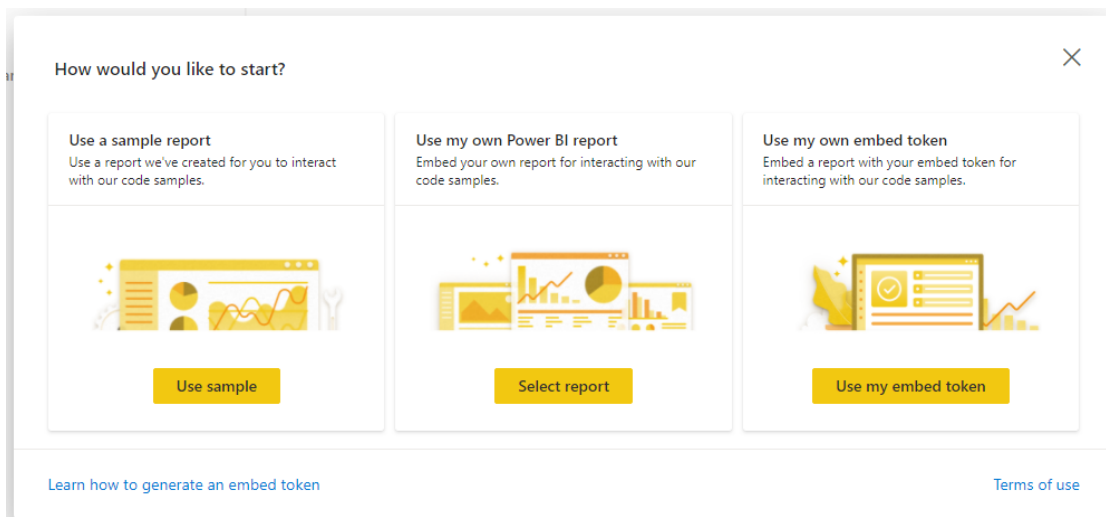
## Updated Multi-Geo support for Power BI Embedded

Multi-Geo support for Power BI Embedded means that anyone that builds applications using Power BI Embedded to embed analytics into their apps may deploy their data in different regions around the world. Until now, Multi-Geo did not result in better performance for

Power BI Embedded deployments, as loading reports and dashboards still involved data requests to the home region. This update overcomes this limitation, meaning data requests are sent directly to the capacity, resulting in a much better performance.

## Power BI Embedded Playground update: use your own embed token

You can now embed a report in the playground developer sandbox by using your own embed token. This is in addition to the existing options of embedding a sample report or selecting one of your available Power BI reports.



If you're using the 'embed for your customers' solution, you'll be generating embed tokens using REST APIs to let your app know which Power BI content can be accessed by its users and what their access level is.

Once you've generated an embed token, you can enter it in the playground to test the report and ensure that the correct content and

access levels have been set. You can then continue to explore our APIs and test them out on your report.

If you're using row-level security (RLS) to restrict user access to data, you can test it in the developer sandbox by inserting an embed token generated for the relevant identity you would like to test.

## Power BI Angular component

The Power BI component for Angular was recently released. This component lets you easily embed your Power BI reports, dashboards and more into your Angular web application.

Over the past few years, the popularity of Angular has greatly increased, and it is currently one of the most-used web frameworks by professional developers. The new Power BI component for Angular gets you from insights to actions quickly and easily, by providing the needed framework

for integrating Power BI analytics in your Angular app.

Using this library, you can embed Power BI reports, dashboards, tiles, visuals, Q&A and paginated reports. You can also optimise your report's performance using the available bootstrap integration or phased embedding, set and edit event handlers, and take advantage of the available client-side APIs.

## Add modern ToolTip to your custom visual

Now you can add a modern ToolTip to your custom visuals with API V3.8.3 release or higher, by adding the supportEnhancedTooltips property to the tooltips object in the capabilities.json file.

## Modal Dialog API update

The new modal dialog API gives developers the ability to determine the size and the position of the dialog relative to the originating visual. This is available with API 4.0 release.

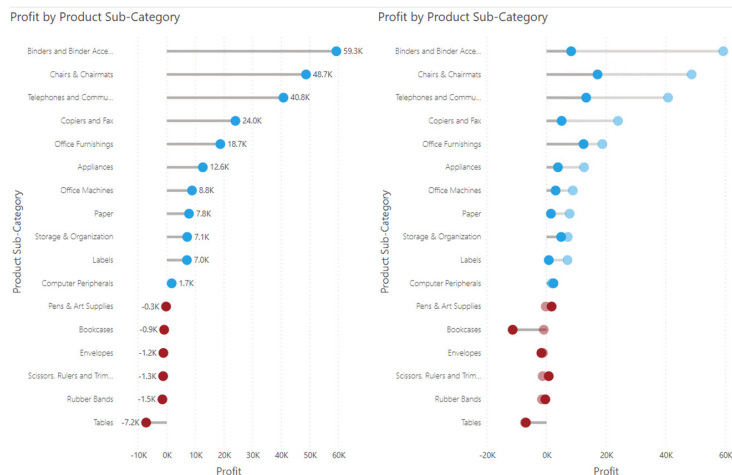
## Lollipop Bar Chart by Nova Silva

Standard bar charts are ideal for showing a single measure per category. You can easily compare each of the categories with the rest. However, if you have a larger number of categories say, more than 10) in a bar chart, it's possible the chart itself becomes "heavy". The coloured bars will fill a large part of the chart surface. To avoid this clutter, you can use the

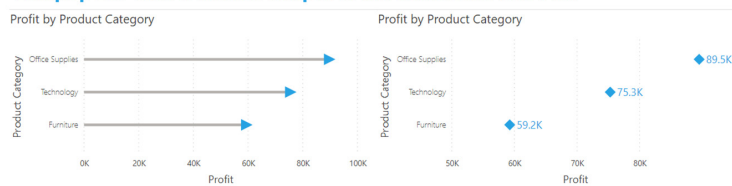
Lollipop Bar Chart for Power BI as an alternative.

The Lollipop Bar Chart shows a marker (usually a dot) per category. A subtle line connects the marker to the measure-axis origin. The marker combined with the line make it a Lollipop Bar Chart.

### Lollipop Bar Chart: Data Labels, Conditional Formatting & Partial Highlighting



### Lollipop Bar Chart: Marker Shapes & the additional Dot Plot



The Lollipop Bar Chart also contains an additional chart: the dot plot. All you have to do is hide the line connecting the marker to the axis.

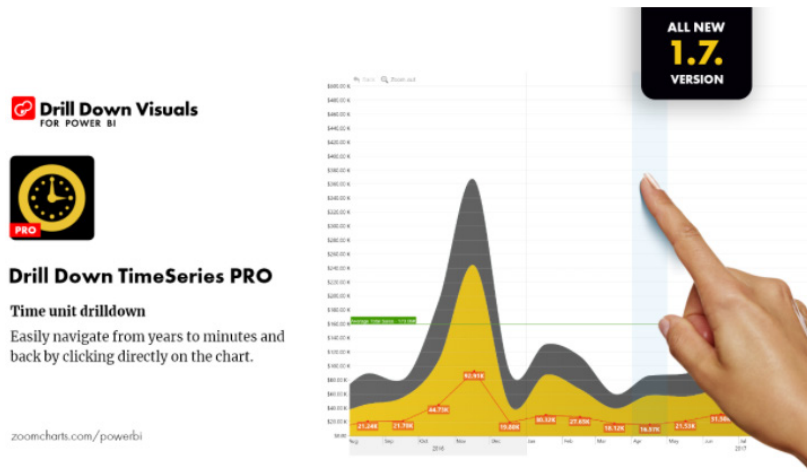
The latest version of this chart contains the following enhancements:

- conditional formatting (both the marker and the line)
- partial highlighting
- data labels
- marker shapes.

The Lollipop Bar Chart can be downloaded from the AppSource. All features are available for free for evaluation purposes within Power BI Desktop.



## Drill Down TimeSeries PRO visual by ZoomCharts



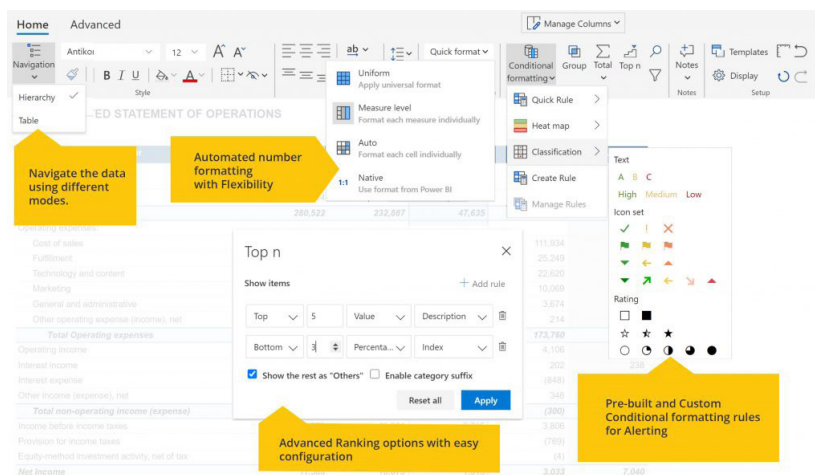
You can visualise a Power BI timeline with Drill Down TimeSeries PRO by ZoomCharts. This visualisation lets you explore time-based data right down to the millisecond. It features numerous interaction options and smooth animations, combined with multiple chart types. Periodicity may be switched from months to weeks or days, in order to drill down into the data, and choose between panning and selecting as your filtering method.

Among the main features of Drill Down TimeSeries PRO you will find:

- multi-touch device friendly, with similar user experience on any device
- multiple chart types and combinations: column, area and line charts. This includes stacking and clustering for all chart types
- cross-chart filtering: instead of using slicers, select data points on multiple charts
- static and dynamic thresholds: set up multiple thresholds to demonstrate targets or benchmarks
- full customisation: you may select various options to combine multiple chart types, set thresholds, and convert GMT data to local time, etc.

You can evaluate the Power BI timeline capabilities of Drill Down TimeSeries PRO for free, with a 30-day trial from the AppSource.

## Inforiver by Lumel



Inforiver is an innovative Low code / No code reporting and analytics solution for Microsoft Power BI. End-users can use Inforiver to rapidly build complex financial statements, P&L reports, management and variance reports, with the help of a built-in visual formula engine with no IT or DAX knowledge.

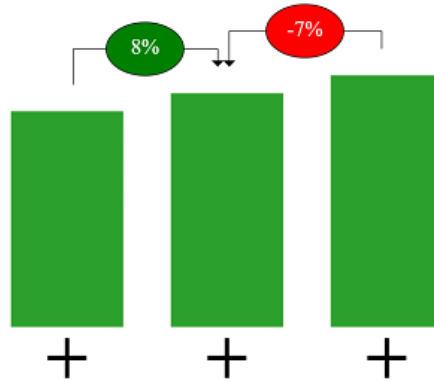
The Inforiver professional edition also enables pivot table type analysis, nested filtering and Top N ranking capabilities. The built-in toolbar provides single-click styling, alignment and several formatting options. Highlights include:

- Financial & management reporting:
  - o Excel-like quick formatting (cell, rows and columns)
  - o pagination
  - o row / column reordering and grouping on the fly
  - o automatic number formatting
  - o grand and subtotal positioning (top or bottom), as well as split totals
  - o insert visual formula row / formulae, as well as static manual input rows
  - o ragged hierarchy handling
- Collaboration:
  - o cell-level notes and annotations
  - o audit: capture visual level changes for changelog
  - o cell edits and cell level formulae
- Analysis use cases:
  - o expand / collapse individual columns for pivot analysis
  - o able to hide /show using manage columns (asymmetric tables)
  - o advanced conditional formatting (totals, values)
  - o freeze and pin: multiple rows at the same time and across pages.

You can download this visual from AppSource.

## Hierarchical bar chart by Excelnaccess.com

Hierarchical bar chart displays hierarchical data (different fields having parent / child relationship(s)) in the form of a bar / column chart with +/- signs to view / hide details or child elements.



Features include:

- expand / collapse bars using (+/-) buttons
- show variance between bars
- drag the bars for custom sorting
- click on legends to drill down/up to any level
- show targets.

You can download this visual from AppSource.

## New visuals in AppSource

Other new visuals this month include:

- Acterys Reporting by Managility
- Drill Down Pie PRO by ZoomCharts
- Floor Plan Visual by Simpson Associates
- Journey Chart by MAQ Software
- Phrazor by vPhrase Analytics Solutions Pvt. Ltd.
- PmBI – Process Mining by ProcessM
- Ultimate Waterfall by dataviz.boutique.

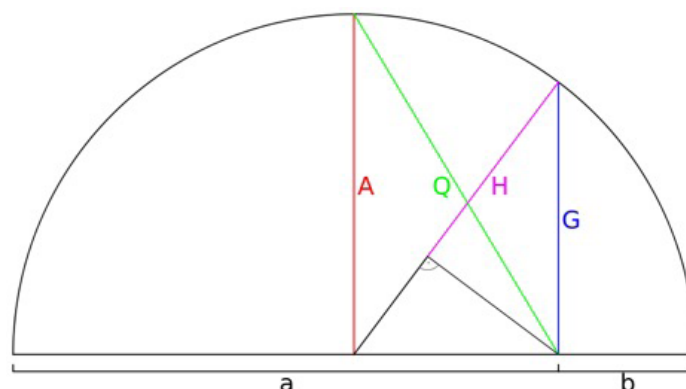
## ODBC support for paginated reports now Generally Available.

ODBC support is now Generally Available for Power BI paginated reports. You may now connect to any data source with an ODBC option when creating a paginated report in Power BI Report Builder. Paginated reports may now connect to ODBC drivers such as SAP Hana, Redshift, Snowflake, Oracle, DB2, Vertica, etc.

More next month, we're sure!

## The A to Z of Excel Functions: HARMEAN

A dangerous function for Ean, they say a picture tells a thousand words:



This time, I think this is true, provided the words are in Swahili (apologies to both of our Swahili readers).

In mathematics, there are three averages known as the **three classical Pythagorean means** are the arithmetic mean (AM), the geometric mean (GM), and the harmonic mean (HM). These means were studied with

proportions by Pythagoreans and later generations of mathematicians because of their importance in geometry and music. Face it, if you work in Excel, you're a budding mathematician!

They are defined by:

$$AM(x_1, \dots, x_n) = \frac{1}{n} (x_1 + \dots + x_n)$$

$$GM(x_1, \dots, x_n) = \sqrt[n]{|x_1 \times \dots \times x_n|}$$

$$HM(x_1, \dots, x_n) = \frac{n}{\frac{1}{x_1} + \dots + \frac{1}{x_n}}$$

Each mean, **M**, has the following properties:

- Value preservation

$$M(x, x, \dots, x) = x$$

- Scalar consistency – also known as first order homogeneity

$$M(bx_1, \dots, bx_n) = bM(x_1, \dots, x_n)$$

- Invariance under exchange (i.e. you may swap the values without effect)

$$M(\dots, x_i, \dots, x_j, \dots) = M(\dots, x_j, \dots, x_i, \dots)$$

- Averaging

$$\min(x_1, \dots, x_n) \leq M(x_1, \dots, x_n) \leq \max(x_1, \dots, x_n)$$

The harmonic and arithmetic means are reciprocal duals of each other for positive arguments:

$$HM\left(\frac{1}{x_1}, \dots, \frac{1}{x_n}\right) = \frac{1}{AM(x_1, \dots, x_n)}$$

whereas the geometric mean is its own reciprocal dual:

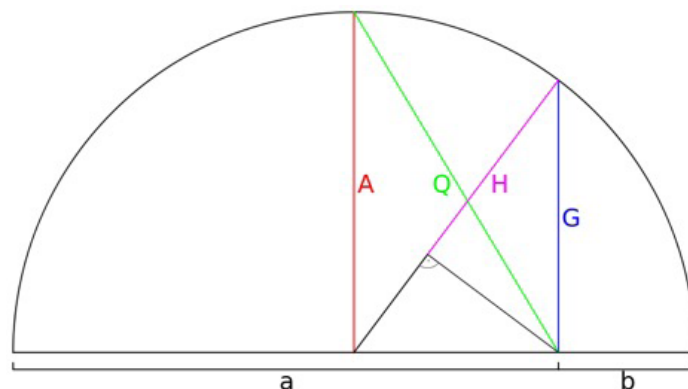
$$GM\left(\frac{1}{x_1}, \dots, \frac{1}{x_n}\right) = \frac{1}{GM(x_1, \dots, x_n)}$$

There is an ordering to these means (if all of the  $x_i$  are positive)

$$\min \leq HM \leq GM \leq AM \leq \max$$

with equality holding if and only if the  $x_i$  are all equal.

Returning to our graphic,



this is a geometric construction of various means of **a** and **b**:

- the **arithmetic average A (AVERAGE)**
- the **geometric mean G (GEOMEAN)**
- the **harmonic mean H (HARMEAN)**
- the **quadratic mean Q** (also known as the root mean square, which is not a Pythagorean mean, but is “simply” the square root of the arithmetic mean of the squares of a set of numbers). It has no Excel function, but can be calculated using basic operators and functions in Excel.

With understanding, it provides a graphical illustration of the relationship between these four means.

The harmonic mean (also known as the subcontrary mean) is often used to calculate average rates, for example. It is simple to calculate:

$$\left( \frac{1^{-1} + 4^{-1} + 4^{-1}}{3} \right)^{-1} = \frac{3}{\frac{1}{1} + \frac{1}{4} + \frac{1}{4}} = \frac{3}{1.5} = 2$$

The **HARMEAN** function has the following syntax:

**HARMEAN(number1, [number2], ...)**

It has the following argument(s):

- **number1, number2, ...**: only **number1** is required, subsequent numbers are optional. You may have between one (1) and 255 arguments to calculate the mean
- you may also use a single array or a reference to an array instead of arguments separated by commas.

It should also be noted that:

- as stated above, the harmonic mean is always less than the geometric mean, which is always less than the arithmetic mean
- arguments can either be numbers or names, arrays or references that contain numbers
- logical values and text representations of numbers that you type directly into the list of arguments are counted
- if an array or reference argument contains text, logical values or empty cells, those values are ignored; however, cells with the value zero (0) are included
- arguments that are error values or text that cannot be translated into numbers cause errors
- if any data point  $\leq 0$ , **HARMEAN** returns the **#NUM!** error value.

Please see our example below:

	A	B	C
1	Data		
2	6		
3	7		
4	9		
5	15		
6	21		
7			
8			
9	Formula	Description	Result
10	<b>=HARMEAN(A2:A6)</b>	Harmonic mean of the data set contained in A2:A6	9.3472
11	<b>=GEOMEAN(A2:A6)</b>	Geometric mean of the data set contained in A2:A6	10.3552
12	<b>=AVERAGE(A2:A6)</b>	Arithmetic mean of the data set contained in A2:A6	11.6000
13			

## The A to Z of Excel Functions: HEX2BIN



*This graphic is misleading – “hex” refers to base 16 – not six [6]!*

This function converts a hexadecimal number (base 16) to binary (base two).

The **HEX2BIN** function employs the following syntax to operate:

**HEX2BIN(number, [places])**

The **HEX2BIN** function has the following arguments:

- **number:** this is required and represents the hexadecimal integer you wish to convert. The **number** cannot contain more than 10 characters. The most significant bit of **number** is the sign bit (40th bit from the right). The remaining 39 bits are magnitude bits. Negative numbers are represented using two's-complement notation
- **places:** this argument is optional. This is the number of characters to use. If **places** is omitted, **HEX2BIN** uses the minimum number of characters necessary. The argument **places** is useful for padding the return value with leading 0s (zeros).

**Two's complement** is a mathematical operation on binary numbers, as well as a binary signed number representation based on this operation. The two's complement of an **N**-bit number is defined as the complement with respect to  $2^N$ ; in other words, it is the result of subtracting the number from  $2^N$ . This is also equivalent to taking the ones' complement and then

adding one, since the sum of a number and its ones' complement is all 1 bit. The two's complement of a number behaves like the negative of the original number in most arithmetic, and positive and negative numbers can coexist in a natural way.

It should be further noted that:

- if **number** is negative, **HEX2BIN** ignores places and returns a 10-character binary number
- if **number** is negative, it cannot be less than FFFFFFFE00
- if **number** is positive, it cannot be greater than 1FF
- if **number** is not a valid hexadecimal number, **HEX2BIN** returns the #NUM! error value
- if **HEX2BIN** requires more than **places** characters, it returns the #NUM! error value
- if **places** is not an integer, it is truncated
- if **places** is nonnumeric, **HEX2BIN** returns the #VALUE! error value
- if **places** is negative, **HEX2BIN** returns the #NUM! error value.

Please see my example below:

	A	B	C
1	Formula	Description	Result
2	<code>=HEX2BIN("F",8)</code>	Converts hexadecimal F to binary, with eight (8) characters (four [4] zeros at beginning are "padding")	00001111
3	<code>=HEX2BIN("B7")</code>	Converts hexadecimal B7 (not the cell B7) to binary	10110111
4	<code>=HEX2BIN("FFFFFFFF")</code>	Converts hexadecimal FFFFFFFF to binary	1111111111
5			

## The A to Z of Excel Functions: HEX2DEC



This function converts a hexadecimal number (base 16) to decimal (base 10).

The **HEX2DEC** function employs the following syntax to operate:

**HEX2DEC(number)**

The **HEX2DEC** function has the following argument:

- **number**: this is required and represents the hexadecimal integer you wish to convert. The **number** cannot contain more than 10 characters (40 bits). The most significant bit of **number** is the sign bit (40th bit from the right). The remaining 39 bits are magnitude bits. Negative numbers are represented using two's-complement notation.

**Two's complement** is the same mathematical operation as described in **HEX2BIN** (above).

It should be further noted that:

- if **number** is not a valid hexadecimal number, **HEX2DEC** returns the **#NUM!** error value.

Please see examples below:

	A	B	C
1	Formula	Description	Result
2	<b>=HEX2DEC("A5")</b>	Converts hexadecimal A5 (not the cell A5) to decimal	<b>165</b>
3	<b>=HEX2DEC("FFFFFFF5B")</b>	Converts hexadecimal FFFFFFF5B to decimal	<b>-165</b>
4	<b>=HEX2DEC("3DA408B9")</b>	Converts hexadecimal 3DA408B9 to decimal	<b>1034160313</b>
5			

## The A to Z of Excel Functions: HEX2OCT



This function converts a hexadecimal number (base 16) to octal (base eight).

The **HEX2OCT** function employs the following syntax to operate:

**HEX2OCT(number, [places])**

The **HEX2OCT** function has the following arguments:

- **number**: this is required and represents the hexadecimal integer you wish to convert. The **number** cannot contain more than 10 characters. The most significant bit of **number** is the sign bit (40th bit from the right). The remaining 39 bits are magnitude bits. Negative numbers are represented using two's-complement notation
- **places**: this argument is optional. This is the number of characters to use. If **places** is omitted, **HEX2OCT** uses the minimum number of characters necessary. The argument **places** is useful for padding the return value with leading 0s (zeros).

**Two's complement** is blah, blah, blah... *again*.

It should be further noted that:

- if **number** is negative, **HEX2OCT** ignores **places** and returns a 10-character octal number
- if **number** is negative, it cannot be less than FFE000000
- if **number** is positive, it cannot be greater than 1FFFFFFF
- if **number** is not a valid hexadecimal number, **HEX2OCT** returns the **#NUM!** error value

- if **HEX2OCT** requires more than **places** characters, it returns the **#NUM!** error value
- if **places** is not an integer, it is truncated
- if **places** is nonnumeric, **HEX2OCT** returns the **#VALUE!** error value
- if **places** is negative, **HEX2OCT** returns the **#NUM!** error value.

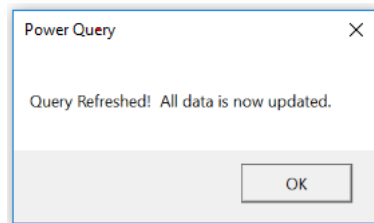
Please see our final example for this month below:

	A	B	C
1	Formula	Description	Result
2	=HEX2OCT("F",3)	Converts hexadecimal F to octal with three characters (the leading 0 is "padding")	017
3	=HEX2OCT("3B4E")	Converts hexadecimal 3B4E to octal	35516
4	=HEX2OCT("FFFFFFFF00")	Converts hexadecimal FFFFFFFF00 to octal	777777400
5			

More Excel Functions next month.

## Beat the Boredom Suggested Solution

Earlier in this newsletter, we asked how can you create a file that will refresh your Power Query queries, and pop up a message box letting us know when it's finished, without disabling background refresh (and therefore rendering Excel unusable)?



Our solution here takes advantage of not one, but two separate quirks of Power Query and the way it refreshes data. It's also particularly inefficient, and we're hoping that Microsoft will eventually make some tweaks to allow this to be a bit more elegant. Let's explain.

The first quirk is that refreshing queries will trigger calculation updates in Excel. This means that we can use a query update to trigger a calculation event in VBA. So our thought was to create an extra query that would simply contain a single random number using the Number.Random()

function in Power Query. This would create an output table that would contain a number that changes whenever the query is refreshed. We can set up a VBA event to check to see if the number has changed. This required a few steps: set up a public variable that can be updated, assign the variable a starting value when the model is opened (to ensure that it doesn't pop up a message when any calculation is performed) and then run a check to see whenever the value changes.

```
Public RandomNumber As Double
```

```
Private Sub Workbook_Open()  
RandomNumber = Range("FinalQuery").Value  
End Sub
```

```
Private Sub Workbook_SheetCalculate(ByVal Sh As Object)  
If Range("FinalQuery").Value = RandomNumber Then  
    'Do Nothing  
Else  
    RandomNumber = Range("FinalQuery").Value  
    MsgBox "Query Refreshed! All data is now updated.", vbOKOnly, "Power Query"  
End If  
End Sub
```

In principle, this should work; in practice, this fails because Power Query will go ahead and run all queries that it can simultaneously. Because the random number query is so simple, it finishes first and triggers the macro

before the other queries may be completed. This brings us to the second quirk that we will take advantage of.

Power Query refresh speeds are not always that quick, to put it politely. There are two things to note:

1. Power Query tends to go back to the data source more frequently than it should in order to refresh tables (we'll come back to this point later)
2. Queries are run in order of hierarchy, i.e. those that rely upon others will be delayed until the precedent queries are run. Queries that are on the same level may be run simultaneously.

Therefore, we should create a query that will reference all of the other outputs, create our random number, then get rid of all the underlying data from the other queries, so that all we're left with is the random number.

Here's the M code that I've used to make it happen:

```
let
    Source = Table.Combine({#"Output Table", #"Output Table 2", #"Output Table 3"}),
    #"Added Custom" = Table.AddColumn(Source, "RandomNumber", each Number.Random()),
    #"Removed Other Columns" = Table.SelectColumns(#"Added Custom",{"RandomNumber"}),
    #"Kept First Rows" = Table.FirstN(#"Removed Other Columns",1)
```

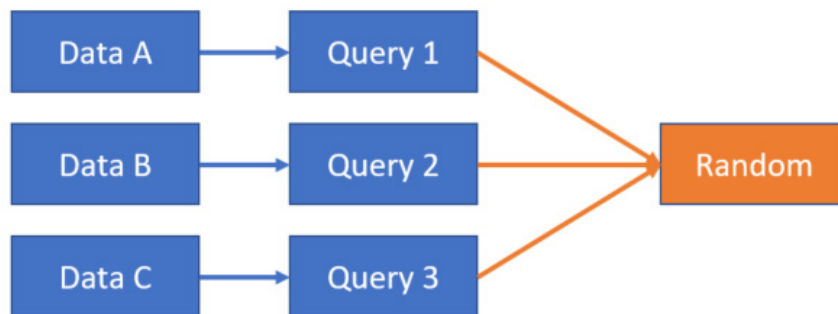
```
in
    #"Kept First Rows"
```

This creates a table in Excel, containing just a single random number. Here's where there's actually yet another quirk to note about query refreshing: for the Calculation event to trigger, the value needs to be a precedent cell in a calculation.

	A	B	C
1	RandomNumber		
2	0.936503487		0.936503487
3			=FinalQuery

Thus, we may force the random number to be the last thing calculated, which will result in our message box above appearing only when all queries have been refreshed. The timing of this might be a little bit random, as it's linked to how long it takes for the query to run, but it should still trigger when it happens. Fortunately, if you're in the middle of entering in a formula in a cell, calculations are intrinsically switched off while you're editing, so you're not going to be interrupted with a pop-up until after you finish with your formula.

So, what's wrong with this? Well, it goes back to the point earlier about the efficiency of Power Query. If you think about what happens in this random query, it's calling upon the three initial queries. Due to the way that Power Query operates at the moment, it's actually triggering another call back to the underlying data, which is effectively going to double the refresh time in this instance.



In the future, I have no doubt that this will be fixed and that the data will be cached into the staging queries accordingly.

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](mailto: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	3 - 5 Nov 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	10 Nov 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	1 Day
Online (Australia)	Financial Modelling	11 - 12 Nov 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	8 - 10 Dec 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	15 Dec 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	1 Day
Online (Australia)	Financial Modelling	16 - 17 Dec 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 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 year, we thought we'd revisit each function key in depth (there are 12 – one for each month of the year!). Given it's now November, let's look at the **F11** tips.

Keystroke	What it does
F11	Insert chart on new worksheet
ALT + F11	Show VBA Editor
CTRL + F11	Insert new Macro worksheet
SHIFT + F11	Insert new worksheet
ALT + SHIFT + F11	Show Script Editor

Hey, it's not our fault so many do the same thing this month!

There are c.550 keyboard shortcuts in Excel. For a comprehensive list, please download our Excel file at [www.sumproduct.com/thought/keyboard-shortcuts](http://www.sumproduct.com/thought/keyboard-shortcuts). Also, check out our new daily **Excel Tip of the Day** feature on the [www.sumproduct.com](http://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](mailto: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](http://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](mailto: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](mailto:training@sumproduct.com) for a copy of the brochure or download it directly from [www.sumproduct.com/training](http://www.sumproduct.com/training).

**Sydney Address:** SumProduct Pty Ltd, Suite 803, Level 8, 276 Pitt Street, Sydney NSW 2000  
**New York Address:** SumProduct Pty Ltd, 48 Wall Street, New York, NY, USA 10005  
**London Address:** SumProduct Pty Ltd, Office 7, 3537 Ludgate Hill, London, EC4M 7JN, UK  
**Melbourne Address:** SumProduct Pty Ltd, Ground Floor, 470 St Kilda Road, Melbourne, VIC 3004  
**Registered Address:** SumProduct Pty Ltd, Level 14, 440 Collins Street, Melbourne, VIC 3000

[contact@sumproduct.com](mailto:contact@sumproduct.com)  
[www.sumproduct.com](http://www.sumproduct.com)  
**+61 3 9020 2071**