Sum Product

We're a little introspective this month as we announce two internal pieces of news (our very own **Steve Kraynak** becomes an Excel MVP and we have a new brochure for your perusal), plus take a personal look through our newest MVP at what it's like to attend the very recent Microsoft MVP Summit (expensive, apparently…).

Of course, we have *most* of our usual features. I say "most" as the Power BI Updates announcement has been released too late for this newsletter, but no mind – let's hope they make a *Swift* return next month...

We still have to take a look at why we love merged cells so much (newsflash: we don't), plus there are the usual Beat the Boredom Challenge, Charts & Dashboards tips, Excel for Mac, Visual Basics, Power Pivot Principles, Power Query Pointers and Excel Updates. We even put our A to Z of Keyboard Shortcuts under the **UV** and we have more Excel functions this **MONTH** too...

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

Liam Bastick, Managing Director, SumProduct

Introducing Our New Brochure

We are excited to launch our new company brochure: SumProduct - Your Partners in Modelling, Excel and Power BI.

This comprehensive guide is designed to showcase our range of consulting and training services, including a complete listing of courses available in various learning formats. Whether you are aiming to enhance your skills in the constantly evolving world of Excel and Power BI or seeking support for a modelling project or customised training solution for your finance team, this brochure outlines our extensive experience and expertise.

Check it out by clicking on the link here. And stay tuned for the upcoming launch of our new website!

Steve Kraynak MVP

We are very pleased to announce US head honcho **Steve Kraynak** has been awarded Microsoft's Most Valuable Professional (MVP) award for Excel. This award recognises exceptional technical community leaders from around the world who voluntarily share their high-quality, real-world expertise with others. Microsoft MVPs are a highly select group of experts representing technology's best and brightest who share a deep commitment to community and a willingness to help others.

Worldwide, there are over 100 million participants in technical communities; of these participants, there are c.3,200 active Microsoft MVPs. In Excel, we believe there are approximately 130 that have received this award – and we presently have three of them here at SumProduct!

We certainly think Steve deserves it. After more than a decade as a Product Manager for the Excel team at Microsoft, where he directly impacted many features in Excel, including new worksheet functions like **IFS** and **TEXTJOIN**, Data from Picture, Excel for Mac, user experience improvements like Smooth Scrolling, and many others, Steve has continued to share his enthusiasm and expertise on all things PivotTable, Power Query and Mac Excel on both LinkedIn and on this very website – check out his latest Friday blog on Excel for Mac here.

Welcome to the madhouse, Steve! Hopefully, **Tim Heng** and **Liam Bastick** can rejoin you in July!

Global MVP Summit: A Personal Perspective from New MVP Steve Kraynak

As mentioned above, Steve Kraynak has just become an Excel Most Valuable Professional (MVP). Again, congratulations Steve! As a result – albeit at the last minute – he was invited to the Microsoft MVP Summit at the Microsoft Redmond Campus in Washington state, USA (Tuesday 12 – Thursday 14 March). Yours truly couldn't go this year, so we asked Steve to provide a personal review (below).

Whilst Steve cannot provide any details regarding content (the details revealed to MVPs remains confidential), there was plenty else to remark upon. Over to you, Steve…

I had a very rare opportunity in early March, as a newly minted Microsoft MVP, to attend my first Global MVP Summit as a guest, less than one year after leaving the Microsoft Excel engineering team. I've been on the hosting side for a dozen Global MVP Summits, and now I was a guest! Let me tell you all about my experience in this new role.

What's a Microsoft MVP?

In case you're not aware, MVP stands for Most Valuable Professional. The MVP Award is Microsoft's way of saying "Thanks!" to outstanding community leaders, who are recognised as experts with regard to Microsoft's products. In their own words, "the contributions MVPs make to the community, ranging from speaking engagements to social media posts, to writing books, and to helping others in online communities have incredible impact".

*A group of Microsoft MVPs along with members of the Excel team on the Redmond campus. SumProduct's very own **Steve Kraynak** and **Tim Heng** are third and fourth from left in the front row respectively.*

Each year, Microsoft hosts the Global MVP Summit, an exclusive event hosted in Microsoft's global headquarters in Redmond. This is the chance for MVPs and the product teams to come together in-person (and virtually) to share thoughts, ideas, complaints, wishes and experiences with each other. It's also a time to build relationships that sometimes turn into longtime friendships and even business partnerships. Liam, Tim and I can attest to that, given that I met them at past MVP Summits, and now we're in business together with SumProduct.

Global Reach

Excel MVPs are found around the globe, and while many joined the Summit remotely via Microsoft Teams, there were attendees from near and far, representing all continents aside from Antarctica. We determined that the furthest distance travelled to attend the event was over 15,000 kilometres (9,300 miles for our readers in the US or a half hour's drive for our Australian readers).

MVPs from Australia, Bulgaria, Cambodia, France, Romania, Slovenia, the UK, and the US pose in front of a banner containing the names of all current MVPs.

A Subscription for Success

My first MVP Summit was back in 2012, when I had recently joined the Excel engineering team at Microsoft. In those days, the conference was largely a chance for the engineering team members to give the MVPs a preview of what the next version of Excel would look like and what features it would have. Due to the typical three-year release cycle for Microsoft Office, there wasn't much chance for MVPs to influence the design of new features, but that never stopped them from trying. *[Too right! – Ed.]*

As a new member of the engineering team, I was prepped before meeting and presenting to the MVPs that there would be lots of tough questions about the product and I should be ready to just say, "That's great feedback, thank you" when I either didn't have an answer or couldn't share the real answer.

MVPs Steve Kraynak (left) and Tim Heng (right) from SumProduct along with MVP Tom Urtis.

Things changed dramatically (for the better) with the advent of Office 365 (now Microsoft 365) subscriptions, which ushered in the era of monthly updates. With this new and much faster release cycle, the MVP Summits became much more interesting. The engineering team was now able to ask the MVPs for their opinion about ideas long before committing to their plans. This has been a great change in terms of product improvements and innovation, and a big reason has been a significant increase in the impact and influence of the MVPs.

For example, I worked on new Excel functions around the time that **TEXTJOIN** and **IFS** were released. We were able to show our early designs to the MVPs, who provided feedback that directly impacted our plans. We adjusted some aspects of the function that we did release, and we even cancelled our plans for a few functions that didn't seem important after hearing from MVPs. Microsoft was now much more agile than it had been in the time of three-year release cycles.

Inside and Out

There are typically two distinct perspectives at the MVP Summit – one is Microsoft's perspective and one is that of the MVPs. The Excel MVPs want to learn as much as they can about Microsoft's plans for the product, get explanations about decisions that have been made in the past and share their thoughts and suggestions with the product team to influence the future of Excel. They also want to immerse themselves in the amazing MVP community to network and learn from each other.

On the other side, Microsoft wants to gain insights from the MVPs to help make the products better. To do this most effectively, they share lots of information about their thoughts and plans. They want to hear the opinions about all this from the MVPs, who are experts, and also have extensive knowledge about real-world usage of the products.

It's a win-win situation, but it's not without tension, which comes when honest, frank and passionate feedback highlights any shortcomings of

the product. Excel is a powerful, flexible and feature-rich application, which opens it up to lots of different opinions. The MVP Summit plays host to many spirited discussions about major product decisions, as well as minor aspects. For example, should "PivotTables" have been called something less "techy"? Should Tabular or Compact layout be the default for PivotTables? Why haven't Tables been made to work better?

Now residing on the MVP side of the fence made for an interesting experience. I worked on some features that are still in development, but now I'm not in a position to speak on Microsoft's behalf about those. I have the inside scoop that's unusual for an MVP, but I also have the MVP perspective that's unusual for a member of the product team. I found myself answering questions to the other MVPs to help them understand the inner workings of Microsoft and describing what it's like on the outside to my former colleagues. It was a weird experience for sure.

Connecting and Reconnecting

I was anxious about returning to the place where I spent over 10 years of my career. Those years went by fast, but that's a long time and there are many memories to look back on. If you've ever lived in a house for a long time, then moved away and came back to visit the new owners, you probably have a sense for what I was feeling. I was also reconnecting

with MVPs and connecting with new MVPs, since I missed the 2023 summit. My anxiety quickly went away, thanks to a warm welcome from both Microsoft and the MVPs. It felt great to be back in such an amazing community!

Highlights of the Summit

Overall, Artificial Intelligence (AI) was a highlight, discussed by many of the Summit's attendees. If you've been following any news at all for the past 18 months, you know that AI is all the rage. Opinions about ChatGPT, Copilot and AI were rife. It was truly fascinating to hear the thoughts and predictions of so many talented and intelligent people from the tech industry.

Clearly, MVPs learn from each other all the time. There were experts in the room for just about every aspect of Excel, Power Query, Power Pivot, Power BI and many other Microsoft products. We all realised what an amazing opportunity the MVP Summit is to share with each other and learn from the greats. We discussed and learned about how to provide training, create awesome videos, do things faster, use "tables" and dynamic arrays, export and import data, and much more.

Time for Fun!

Surrounding all the serious and technical discussions there was time for connecting on a personal level. Although this group might include a few computer nerds, they know how to have fun.

A group of MVPs attending a mixer held on Microsoft's campus (Steve Kraynak on left, Tim Heng on right).

Here's a sample of the after-work activities:

- Wine, beer and other winning formulae
- "Business meetings" at a popular bar in Bellevue, WA, including trivia night with team names like XLOOKUPS and the Dynamic Arrays
- Enjoying the famous seafood of the area spread across the table, enjoyed with bibs and mallets

MVPs Oz Du Soleil and Cristiano Galvao enjoying the Seattle seafood scene.

- Did I mention wine and beer?
- A bit of culture with the famous Seattle Symphony
- Hiking and sightseeing around the beautiful Seattle area
- Pictures, pictures, pictures

MVP Steve Kraynak of SumProduct waiting for the Seattle Symphony to perform at Benaroya Hall.

The SUMPRODUCT of it all

Multiplying an array of MVPs with the Microsoft Excel product team, Excel topics that everyone was passionate about, and the beautiful weather that surrounded the MVP Summit, resulted in an amazing experience for everyone. I hope to be back for many years to come.

Merged Cells

A couple of weeks ago, someone was trying to extol the virtues of those pesky merged cells:

Hmmm… I am not so convinced. If anyone wants to know how to create merged cells, you've come to the wrong newsletter. To be clear, if I were to list the benefits, I might choose from the following:

- It allows you to format your spreadsheet in new, exciting and unexpected ways
- Makes programming in VBA referencing cells much more fun
- Stops end users relying on PivotTables for basic data analysis
- Creates extra spacing in your charts
- Prevents users from sorting, copying, pasting, deleting, filtering, data validating or moving data (pretty much anything, really) easily
- You don't need to figure out how to widen columns
- It allows you to hide pre-existing data in secondary cells forever

- It prevents end users from using those infernal dynamic arrays (pro tip: always type a "|" in a merged cell and centre it for maximum chance of non-detection)
- Easier to perpetrate frauds using **SUMIF** and **VLOOKUP** (anything that screws up **VLOOKUP** is A OK in my book) or using auditing tools
- Provides hours of fun for standard Excel users who cannot figure out how to locate them
- Can cause hours of annoyance for expert Power Query users who cannot determine what's wrong with their **M** code
- Probably makes Copilot work better in Excel
- For best effect: use merged cells with different numbers of columns and rows in the same column / row.

Get the picture..?

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…

This month, we are pulling inspiration from our consulting work. The problem here relates to how Excel, Power Query and Power Pivot handle a table / Table with similar column names. The premiss here is that we can normally import data from Excel into Power Query, make a couple of data transformations then export the data to Power Pivot to create measures and other calculations with the data table.

Let's just jump right into our example, assume we have the following Table:

Customer ID	Customer Name	Date	Country	Region	Transaction Type	Transaction type	Amount
1004	Harmonic Sonics	5/09/2018	AU	BBE	Payments	Internal	711
1001	Bizuplyz	29/10/2019	AU	SPA	Payments	External	4,459
1003	L. R. Repeat	25/12/2017	AU	JUY	Payments	Internal	644
1003	L. R. Repeat	20/11/2018	AU	SPA	Payments	Internal	2,472
1003	L. R. Repeat	8/03/2019	AU	JUY	Payments	Internal	5,917
1004	Harmonic Sonics	13/08/2018	AU	SPA	Payments	External	2,352
1003	L. R. Repeat	8/05/2020	AU	GFY	Payments	Internal	508
1003	L. R. Repeat	12/05/2020	AU	SPA	Payments	External	3,749
1004	Harmonic Sonics	29/06/2018	AU	GFY	Payments	External	2,764
1002	Plumb'n'Stuff	22/02/2019	AU	SPA	Payments	Internal	675
1002	Plumb'n'Stuff	7/06/2019	AU	GFY	Payments	External	5,716
1004	Harmonic Sonics	23/01/2019	AU	JUY	Payments	External	756
1002	Plumb'n'Stuff	2/12/2018	AU	NHJ	Payments	Internal	2,543
1004	Harmonic Sonics	10/07/2018	AU	GFY	Payments	External	5,516
1002	Plumb'n'Stuff	17/03/2020	AU	NHJ	Payments	External	3,989

We import this into Power Query:
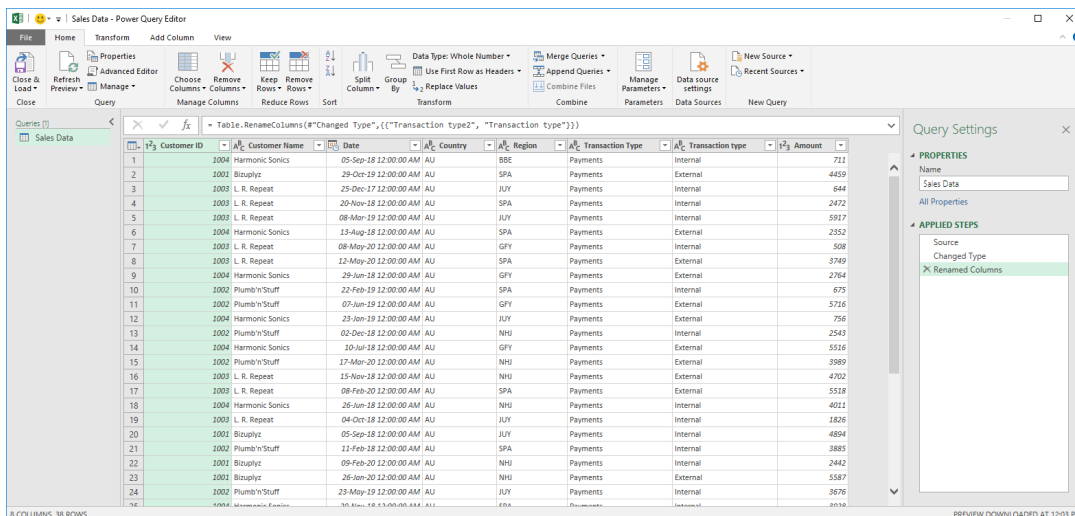
Now, let's imagine we wish to export the data from Power Query into a table in Excel:

Customer ID	Customer Name	Date	Country	Region	Transaction Type	Transaction type2	Amount
1004	Harmonic Sonics	05-09-18 0:00	AU	BBE	Payments	Internal	711
1001	Bizuplyz	29-10-19 0:00	AU	SPA	Payments	External	4459
1003	L. R. Repeat	25-12-17 0:00	AU	JUY	Payments	Internal	644
1003	L. R. Repeat	20-11-18 0:00	AU	SPA	Payments	Internal	2472
1003	L. R. Repeat	08-03-19 0:00	AU	JUY	Payments	Internal	5917
1004	Harmonic Sonics	13-08-18 0:00	AU	SPA	Payments	External	2352
1003	L. R. Repeat	08-05-20 0:00	AU	GFY	Payments	Internal	508
1003	L. R. Repeat	12-05-20 0:00	AU	SPA	Payments	External	3749
1004	Harmonic Sonics	29-06-18 0:00	AU	GFY	Payments	External	2764
1002	Plumb'n'Stuff	22-02-19 0:00	AU	SPA	Payments	Internal	675
1002	Plumb'n'Stuff	07-06-19 0:00	AU	GFY	Payments	External	5716
1004	Harmonic Sonics	23-01-19 0:00	AU	JUY	Payments	External	756
1002	Plumb'n'Stuff	02-12-18 0:00	AU	NHJ	Payments	Internal	2543
1004	Harmonic Sonics	10-07-18 0:00	AU	GFY	Payments	External	5516
1002	Plumb'n'Stuff	17-03-20 0:00	AU	NHJ	Payments	External	3989
1003	L. R. Repeat	15-11-18 0:00	AU	NHJ	Payments	External	4702
1003	L. R. Repeat	08-02-20 0:00	AU	SPA	Payments	External	5518
1004	Harmonic Sonics	26-06-18 0:00	AU	NHJ	Payments	Internal	4011
1003	L. R. Repeat	04-10-18 0:00	AU	JUY	Payments	Internal	1826
1001	Bizuplyz	05-09-18 0:00	AU	JUY	Payments	Internal	4894
1002	Plumb'n'Stuff	11-02-18 0:00	AU	SPA	Payments	Internal	3885
1001	Bizuplyz	09-02-20 0:00	AU	NHJ	Payments	Internal	2442
1001	Bizuplyz	26-01-20 0:00	AU	NHJ	Payments	External	5587
1002	Plumb'n'Stuff	23-05-19 0:00	AU	JUY	Payments	Internal	3676
1004	Harmonic Sonics	29-11-18 0:00	AU	SPA	Payments	Internal	3928
1001	Bizuplyz	12-11-18 0:00	AU	JUY	Payments	Internal	5762

… no problems here so far.

Now let's try to add the data into the data model in Power Pivot. Excel returns with this error message:
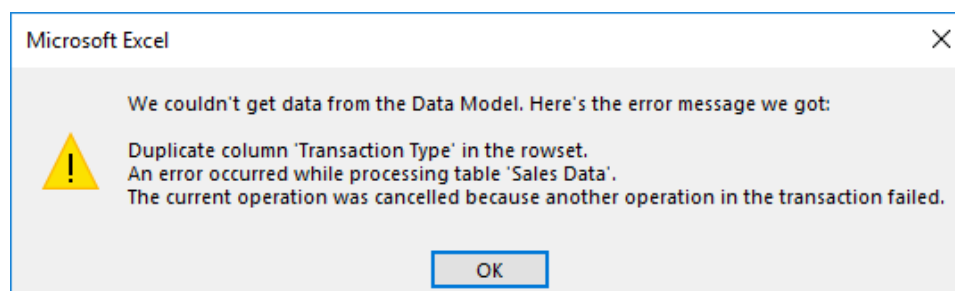
This leaves us in a sticky a tricky situation as our data table did not have any duplicate columns, did it? So how do we go about fixing this? This is the challenge for this month: can you find a solution that will allow you to circumvent this error in Excel when adding the table above into the data model in Power Pivot?

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

Excel for Mac

With Steve Kraynak joining the team, we thought we would exploit his knowledge and recant all about Microsoft Excel for Mac. Each month, we'll cover a different topic to help you understand how Excel for Mac is different than Excel for Windows. This month, it's all about Power Query. For a long time, it didn't even exist on Mac, but since it's such a key feature of Excel, the team at Microsoft has been steadily introducing more and more Power Query capability to Excel for Mac. As at the time of writing, it's still missing a few features, but it at least has a good foundation.

Step 1 of the journey: Refresh

Which came first, the query or the refresh? You can't refresh unless there's a query, right? That's correct, but in Excel for Mac, the ability to refresh came first. This meant the query had to be created in Excel for Windows, and then you could refresh it on Mac.

If you've been using Excel for Mac for a long time, you might be aware that it didn't have any Power Query features until mid-2019. At that time, Microsoft added the ability to refresh queries that were connected to text files. It seems like a very small step forward, but this was a big milestone, because it meant that the foundation of Power Query was there and working.
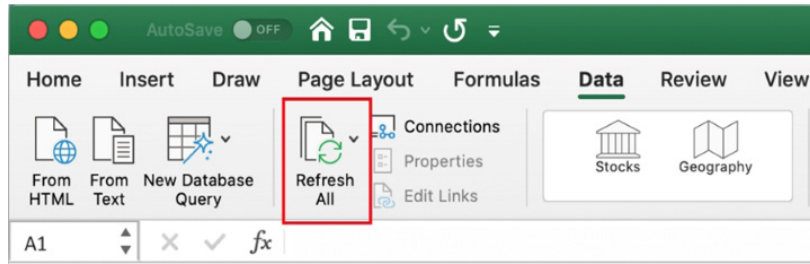
In 2020, you were able to refresh from more types of data source, including tables / Tables within the workbook, SQL Server databases and the first cloud data import capability, with support for files on SharePoint, SharePoint Lists, SharePoint Folder and OData. While it didn't enable much more capability, it showed a big foundational step for Power Query. The ability to sign into online sources is important so that other data sources can be supported in the future.
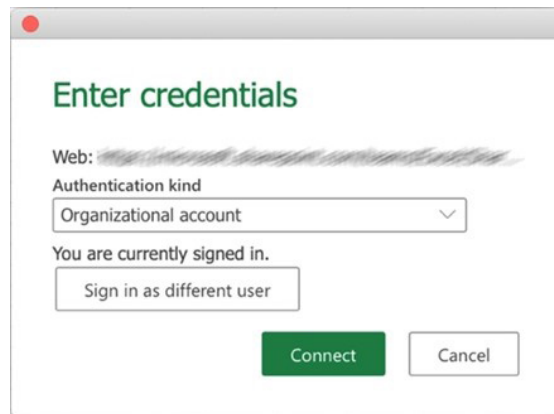
Step 2: 'Get Data'

The next big step came in 2021, when the ability to 'Get Data' was introduced. You could now create new queries from local files to begin importing data from Excel files and text files. Previously, you could refresh queries that were created on Windows, but there was no way to start a new query. The only way to edit a query was by using a VBA workaround, which was not convenient.

Step 3: Transform

In 2022, Microsoft announced that you could not only create new queries, but you could edit them as well with the Power Query editor on Mac. With this update, you could truly 'Get and Transform' your data. This was a major improvement that really brought Excel for Mac to a level where you can get some serious work done.
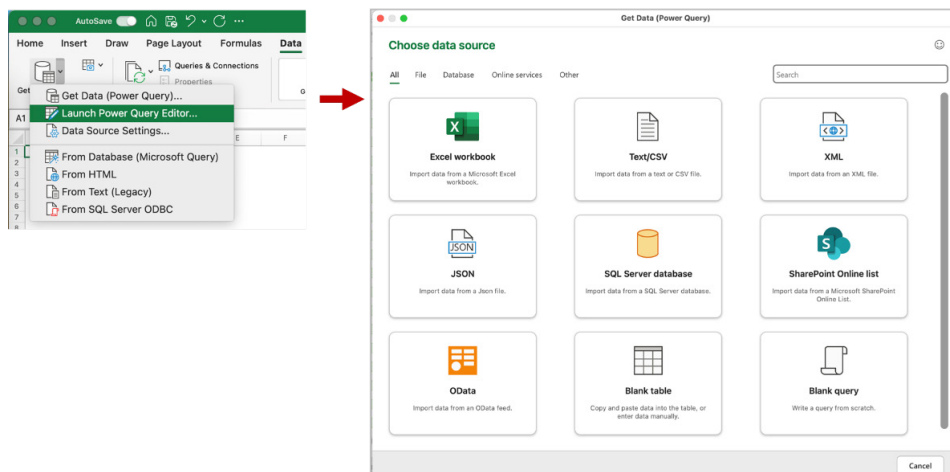
Prior to this improvement, if you picked a data source for a new query, there was no ability to do the Transform steps that makes Power Query so powerful. Now, the full query editor is available on Mac, which is an amazing, so you can transform your data, even if you don't have access to a Windows computer.
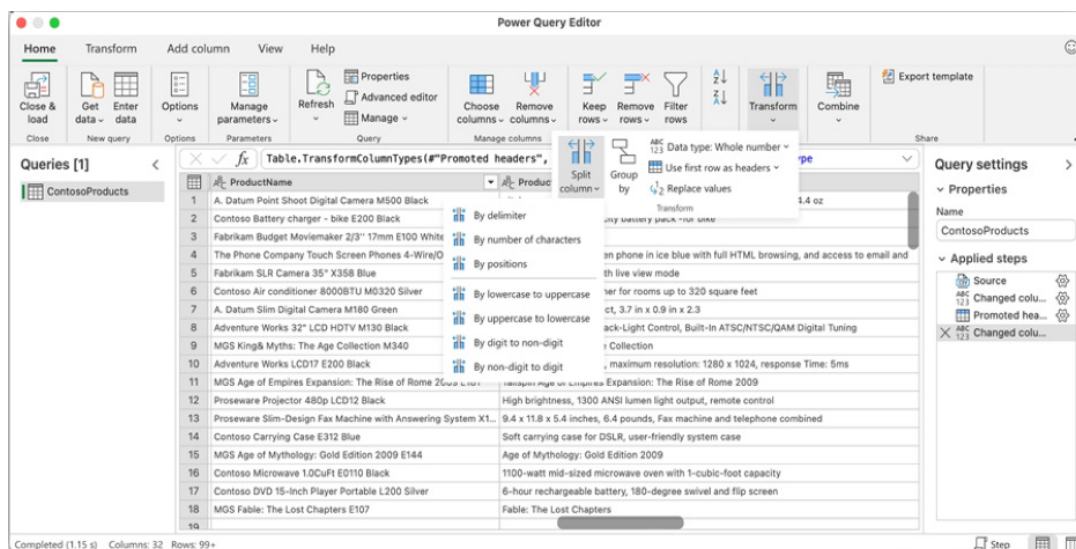
Remaining Limitations

Although creating and editing queries makes Power Query on Mac very capable, there are still a few limitations (as of this writing) that you should be aware of:

- Some data sources are not supported. If you look at the 'Get Data' menu in Excel for Windows, you'll notice a long list of data sources, many of which are not available on Mac. One of the key sources is 'From Table/Range', but there are other alternatives. The full list of data sources supported by different versions of Excel is published by Microsoft here: Power Query data sources in Excel versions

- You can't control the destination of the data from your queries, because the 'Load to…' dialog isn't available on Mac. This means that all queries will get loaded into Tables that appear in your worksheets

- Even if the 'Load to…' dialog were available, you wouldn't be able to add your queries to the Data Model since Power Pivot isn't available on Mac.

Since the Power Query features have been gradually introduced to Excel for Mac, we expect more features to be added over time, so keep watching for updates on the Excel team's blog: Excel Blog (microsoft.com).

Word to the Wise

For your reference, we've included links to Microsoft's Announcements about Power Query on Mac:

- May 2019: A Journey to Power Query in Excel for Mac (microsoft365.com)
- July 2020: Updates to Power Query in Excel for Mac (microsoft365.com)
- October 2020: Journey to Power Query in Excel for Mac continues (microsoft365.com)
- May 2021: Import data from local files with Power Query in Excel for Mac (microsoft365.com)
- May 2022: Shape data with Power Query Editor in Excel for Mac - Microsoft Community Hub

We hope you find this week's topic helpful. Check back for more details about Excel for Mac and how it's different to Excel for Windows.

We'll continue next month…

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 consider Rolling charts.

A "Rolling" chart is just like a rolling budget: it displays the last **x** months (typically, the past 12 months), but keeps up to date *automatically*. The idea is similar, but not quite the same, as we do not wish to extend the range, simply keep moving the 12 months along the time axis.

To do this, you still create a Table (mine is labelled '**Chart_Data**'):

I then calculate the latest date in the formula by typing **=MAX(** and then highlighting cells **F11:F28** in my example. This gives me the formula

=MAX(Chart_Data[Date])

The great thing about this formula expressed in this way is as the dates extend, the range will update automatically. I use this formula in the final row of a second table (check out cell **L22** in the image below):

To populate the rest of the data in column **L**, in cell **L21** I write the following formula

$$=EDATE(L22,-1)$$

This generates the same day of the month one month earlier. I then copy this formula into cells **L11:L20**. Finally, I use a **LOOKUP** formula to derive the Sales data. For example, the formula in the final period here is

$$=LOOKUP(L22,Chart\_Data)$$

Here, the **LOOKUP** formula will always look for the date in the first column of **Chart_Data** and return the value in the final (second) column. You should note that this chart data table is always 12 rows deep for a 12-month look-back. If you need a rolling budget for a different duration, simply amend the number of rows accordingly. It is this table that populates the chart:

From Left

Date	Sales
1 Jul 20	488
1 Aug 20	452
1 Sep 20	459
1 Oct 20	101
1 Nov 20	290
1 Dec 20	476
1 Jan 21	249
1 Feb 21	291
1 Mar 21	136
1 Apr 21	322
1 May 21	189
1 Jun 21	314

Using Second Table
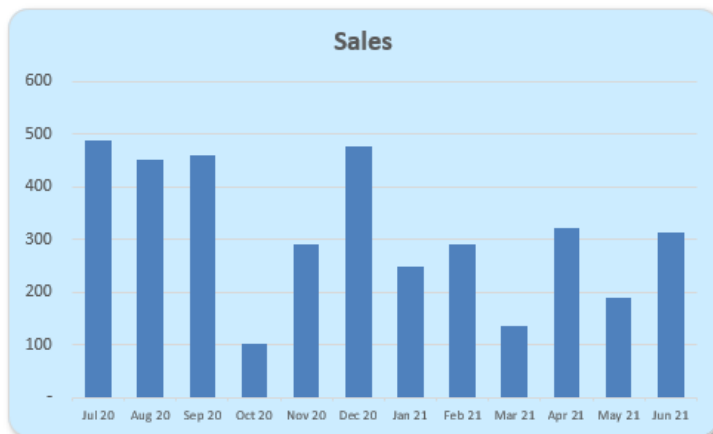
Until next month.

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 consider the concept of a variable scope at a project level.

Variables can also be used in different subroutines and functions. If there are items that are known to have constant values throughout the entire workbook, they can be declared explicitly in one place for easy reference.

Project Level

Sharing variables between procedures within the same module was very simple but what if another module wanted to reuse the same variable?
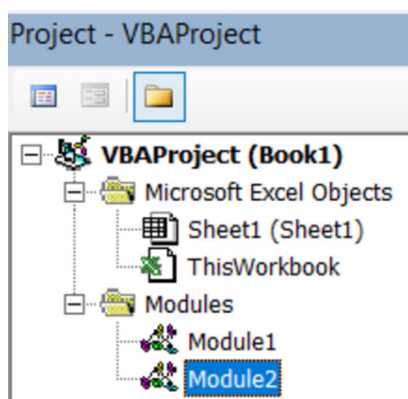
Module2 wants to use **myTestString** but it is declared in **Module1**.
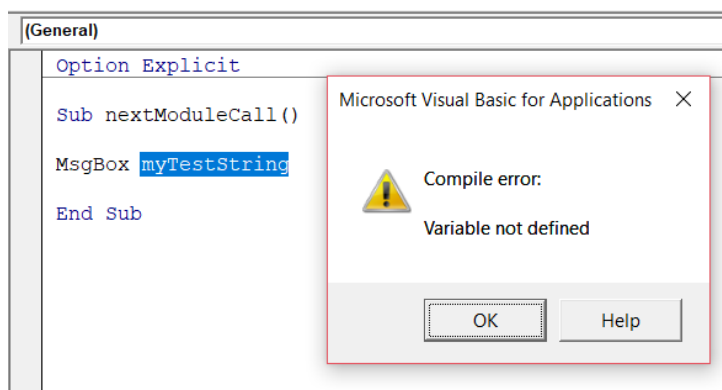
Upon running, it again says the 'Variable not defined'. In order to force a variable to be accessible to the entire workbook the statement, the 'Public' statement may be used.
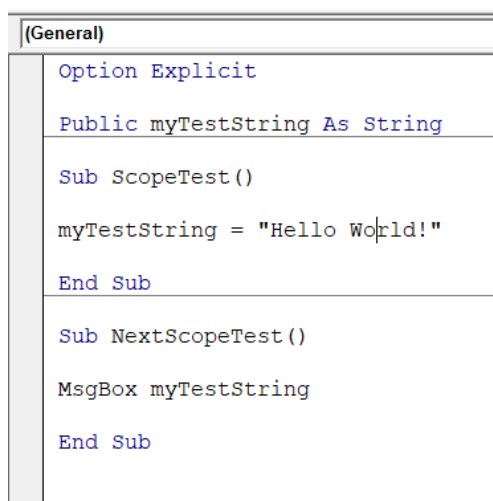
Now this will be accessible from **Module2** so when the following code is called.
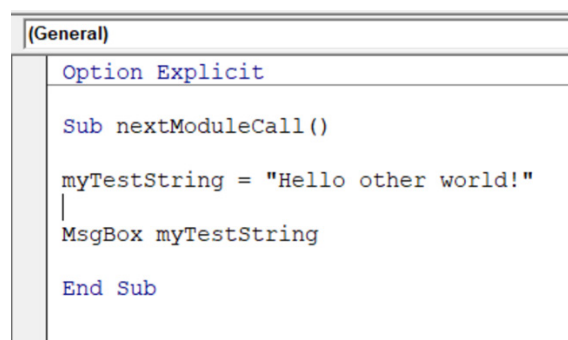
The variable is used as necessary, and the expected result eventuates.
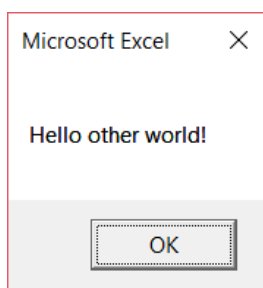
More next time.

Power Pivot Principles

*We continue our series on the Excel COM add-in, Power Pivot. This month, we demonstrate how to use the **LASTNONBLANK** function.*

The **LASTNONBLANK** function returns the last value in the column, where the expression is not blank. This function requires the following syntax to operate:

LASTNONBLANK(column, expression)

The **column** parameter can be a:

- reference to a column
- a table with a single column
- a Boolean expression that defines an individual column.

The **expression** is evaluated on each row of the column for blanks.

In this example, we will be using the following data table:

	A	B	C	D
1				
2		Sales	Category	Date
3		115.00	1	1/06/2019
4		14.00	1	2/06/2019
5		35.00	2	3/06/2019
6		169.00	1	4/06/2019
7		167.00	2	5/06/2019
8		16.00	2	6/06/2019
9		45.00	4	7/06/2019
10		187.00	3	8/06/2019
11		116.00	3	9/06/2019
12		146.00	4	10/06/2019
13				

After we have loaded the Table into Power Pivot and created a PivotTable, we can populate it with some measures:

We first create a measure that sums the total **Sales**:

=SUM(
        SalesAndCategory[Sales]
)

Now, we can create a measure that retrieves the category of the last sale:

=LASTNONBLANK(
        SalesAndCategory[Category],
        [Total Sales]
)
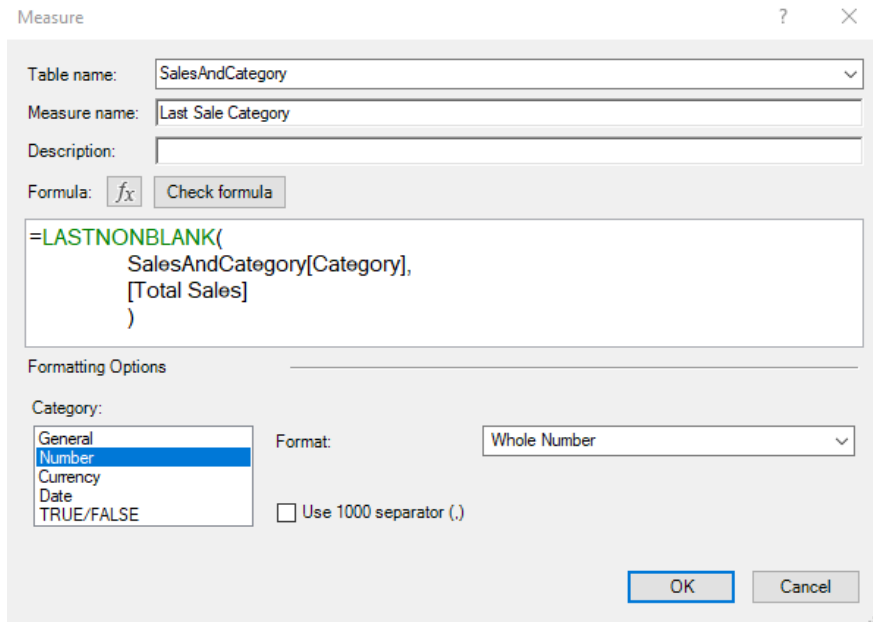
We can see that the measure is in fact returning with category 4:

which is what we would expect.

Now, what if we want a measure that will retrieve the value of the last sale? Following the logic of the previous measure we can write:

=LASTNONBLANK(
        SalesAndCategory[Sales],
        [Sales]
)

Placing the measure in our PivotTable yields:

	A	B	C	D	E	F	G
1							
2		**Sales**	**Category**	**Date**		**Last Sale Category**	**Last Sale Amount**
3		115.00	1	1/06/2019		4	$187.00
4		14.00	1	2/06/2019			
5		35.00	2	3/06/2019			
6		169.00	1	4/06/2019			
7		167.00	2	5/06/2019			
8		16.00	2	6/06/2019			
9		45.00	4	7/06/2019			
10		187.00	3	8/06/2019			
11		116.00	3	9/06/2019			
12		146.00	4	10/06/2019			
13							

That's not quite right. This is because it has arranged the sales in descending order before retrieving the last non-blank value. In order to retrieve the last sale amount based on dates, '$ 146.00', we have to use the **CALCULATE** function:

=CALCULATE(
        [Sum of Sales],
                LASTNONBLANK(
                        SalesAndCategory[Date],
                        [Sum of Sales]
)
)
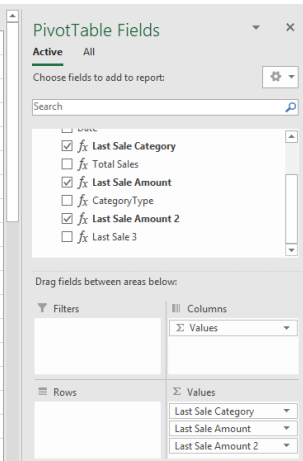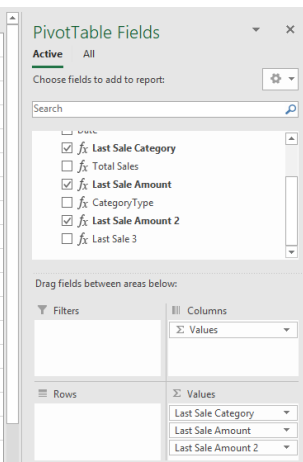
Looking back to our PivotTable,

it now works as expected. These measures are also dynamic: we can insert slicers and have them change accordingly!

Note that if we do not use the **CALCULATE** function with the **LASTNONBLANK** function the measure will arrange the sales in descending order before retrieving the last non-blank value. Don't be caught out!

That's it for this month; more next time.

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 take a look at some tips to help get the best out of Power Query.

Power Query can be used to upload data from a wide range of sources, and it is an excellent tool for cleaning up and formatting data. Here are some ideas that can help along the way to clean and useful data.

1. Quickly rename a column

We often spend a lot of time renaming columns – here is the quickest way.
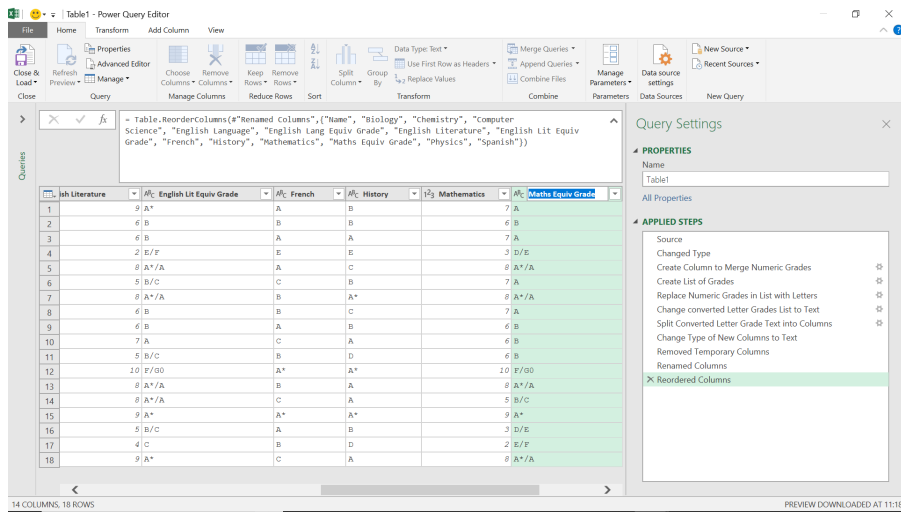
Double left-click on the title of any column, and the default action is to rename a column. Not the most exciting tip, but it saves on mouse clicks.

2. Quickly rename a column without a mouse!

Keyboard shortcuts are always very popular with Excel users! So here's one: to rename a selected column with no clicks, you can use **F2** on the keyboard.
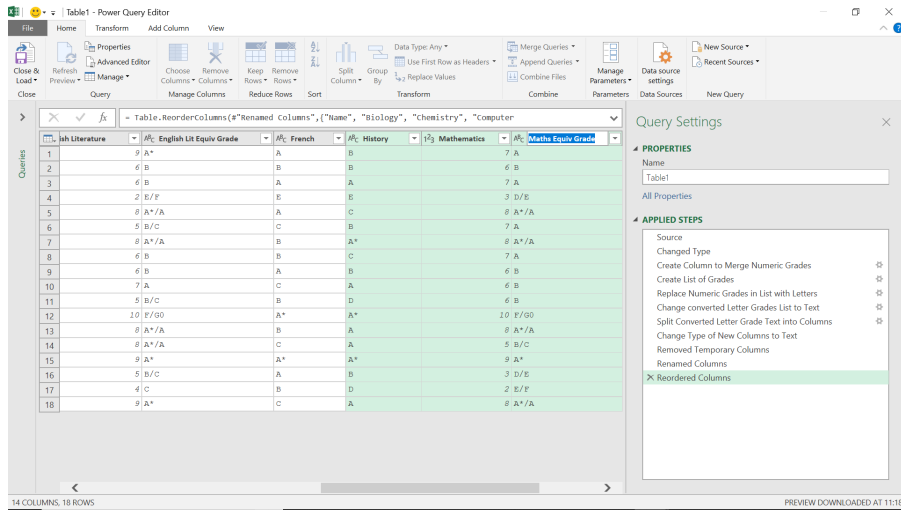
If you have several columns selected, then **F2** will only allow me to rename the last column you selected.

3. Re-write history

I've gone off on something of a tangent in the next example!
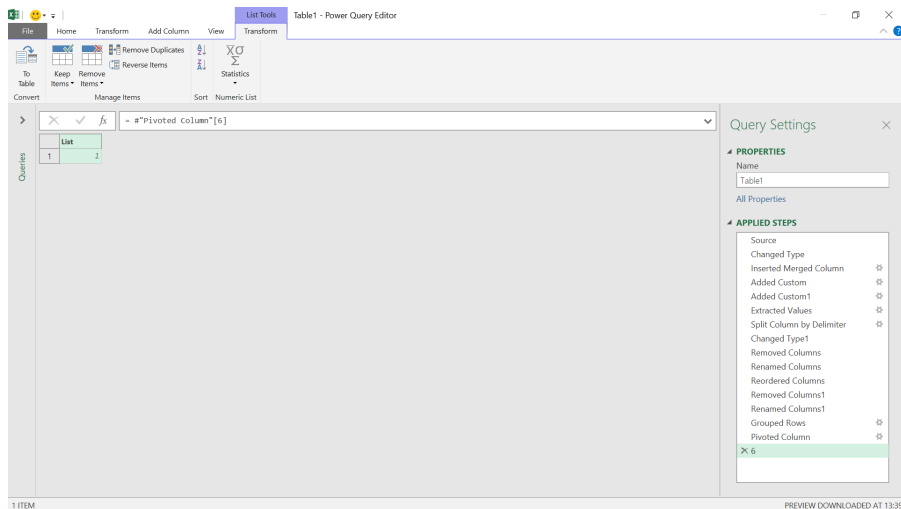
The point where we actually went wrong is at 'Removed Columns1', so we could delete each step in reverse order until we have deleted that step, or we could exit my query and discard the changes if these are the only changes we have made. However, there is an easier way…
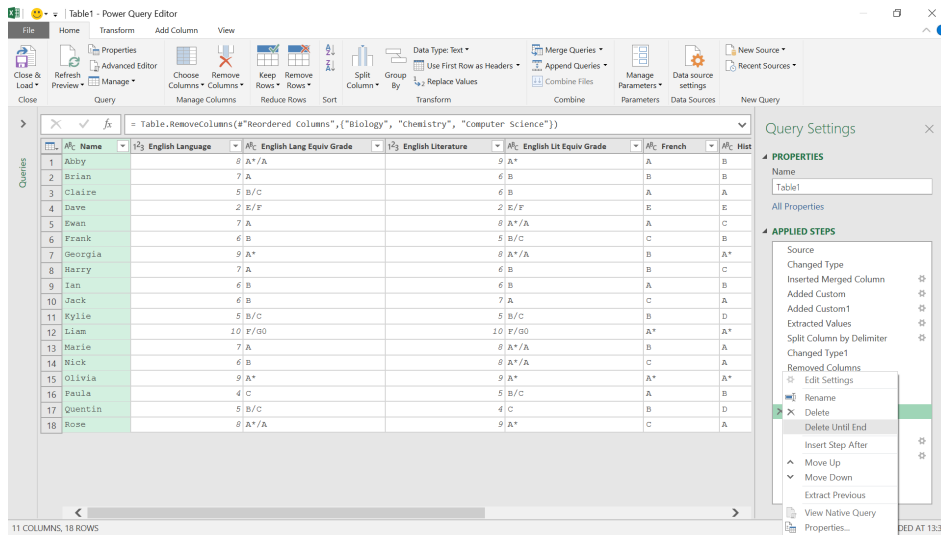
If we select the first step we wish to delete, we can right-click and choose to 'Delete Until End'. This removes everything on my tangent, so we can try again! There is even a check just in case we choose this option by mistake.
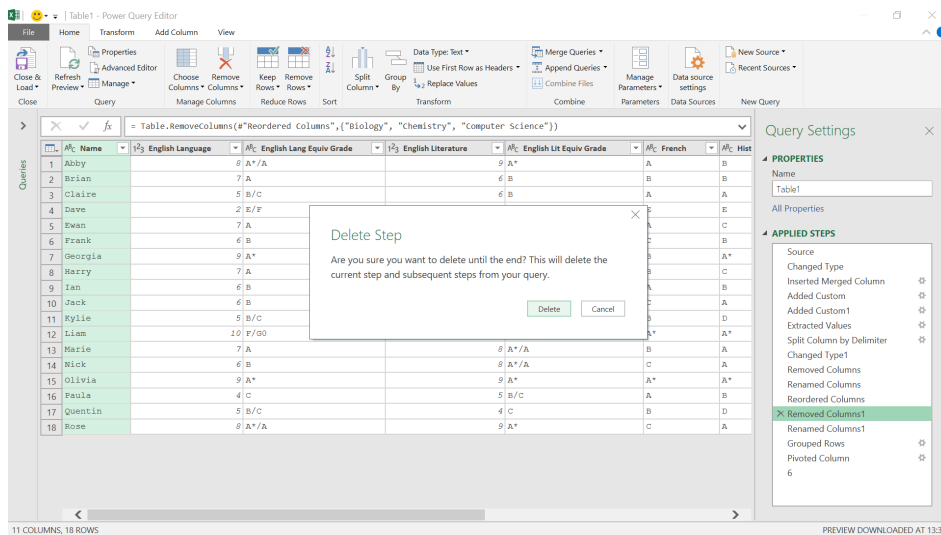
In this case, we want to forget it all happened, so we'll choose 'Delete'.
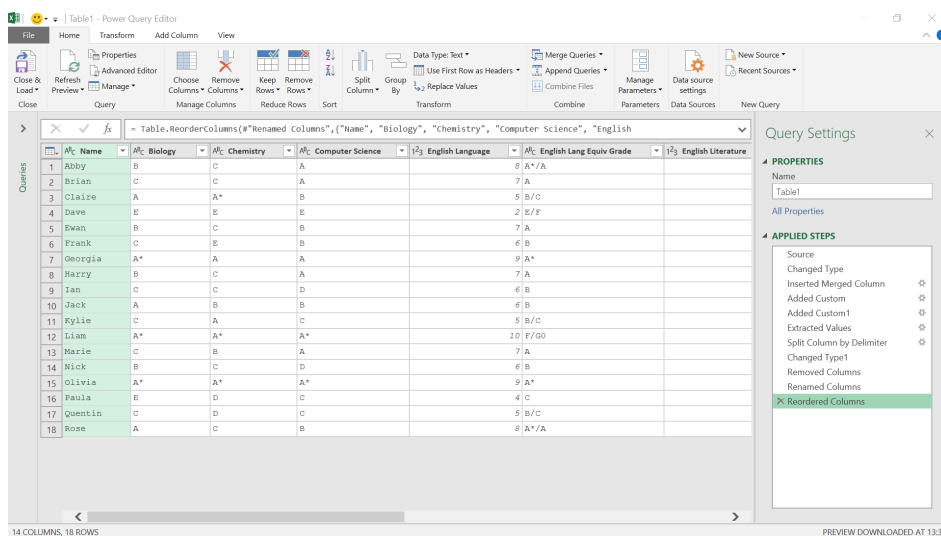
It's as if nothing ever happened…

4. Using monospaced font in the Query Editor

This is useful for when I have references made up of a significant number of characters.
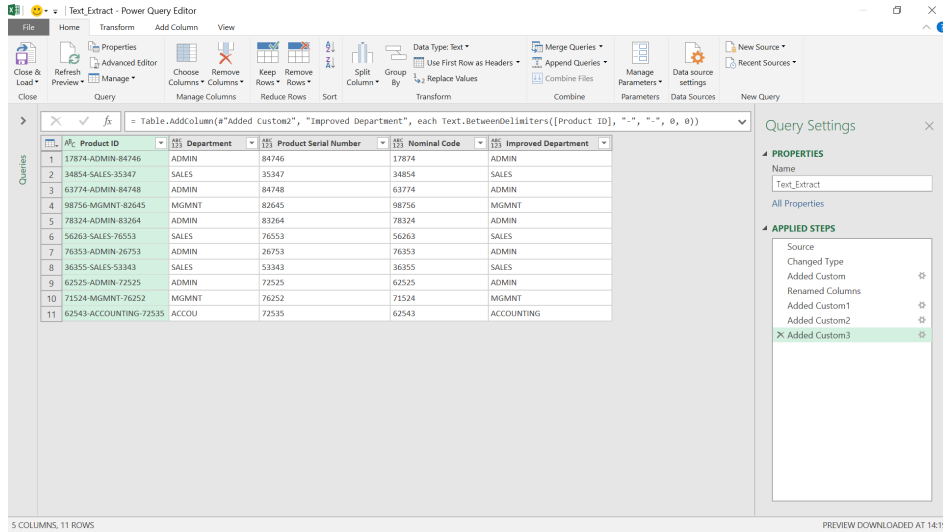
In the previous screen, the Product ID column looks messy. That is because of a setting that we may access from the 'File' tab.
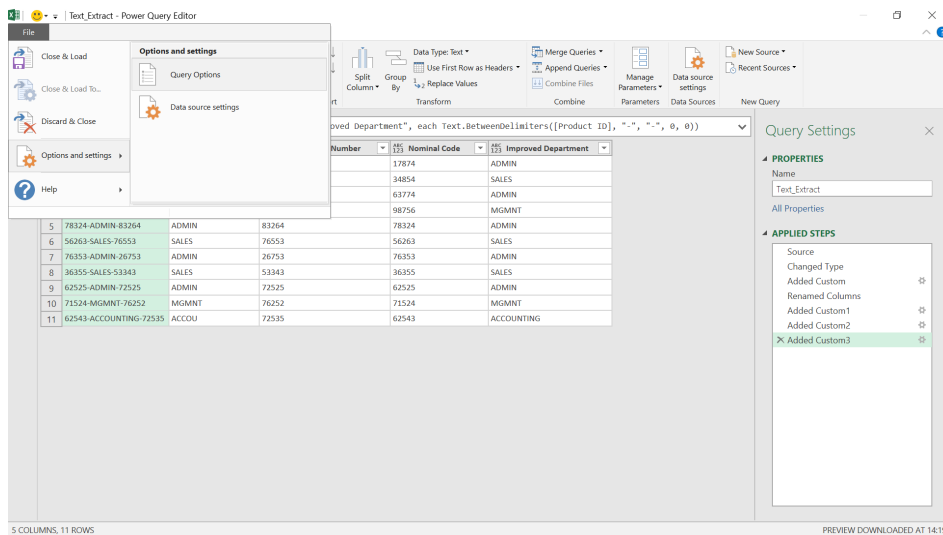
The 'Query Options' available here is much more extensive than the one available from the 'Home' menu.
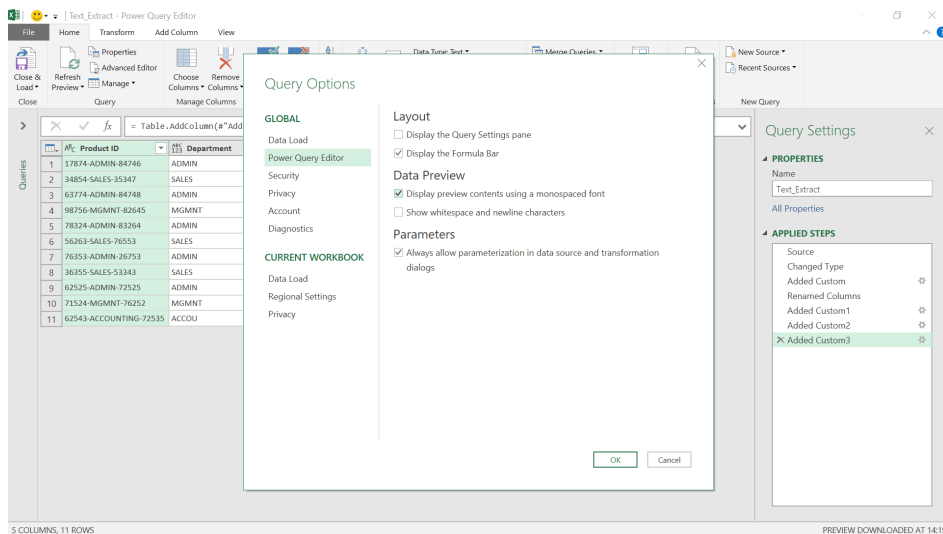
In the 'Power Query Editor' section, we can choose to 'Display preview contents using a monospaced font'. Personally, I always prefer to use this setting.
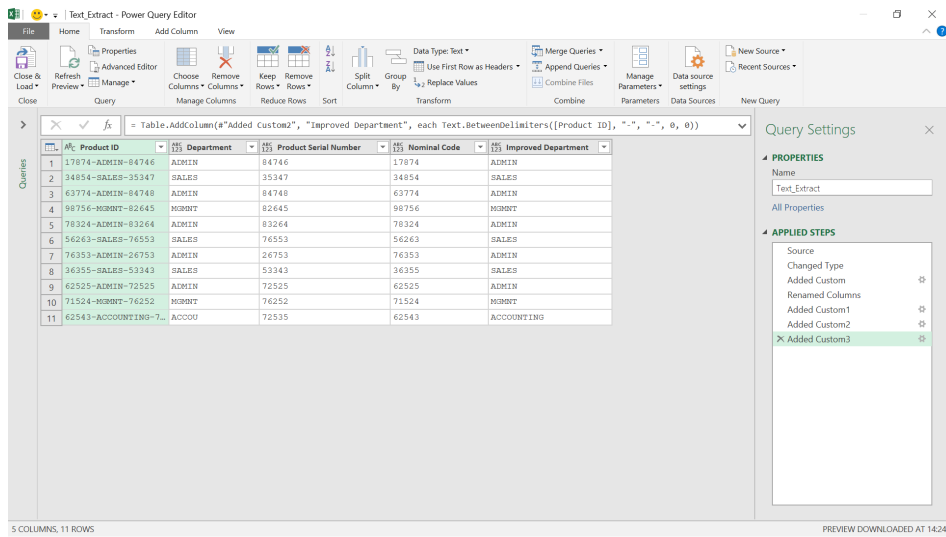
Now when we access the query, the product ID's are neatly aligned (well, except for that pesky last one..).

5. Avoiding the Advanced Editor when entering comments

The Advanced Editor is not the favourite feature of Power Query for many users. After many years of struggling and pleading from users, IntelliSense has now been added to the Power BI **M** editor and is hopefully on its way to 'Get and Transform'. This will improve the editor, but it's still good to know how to avoid it, and it is possible to enter comments from the Formula bar.
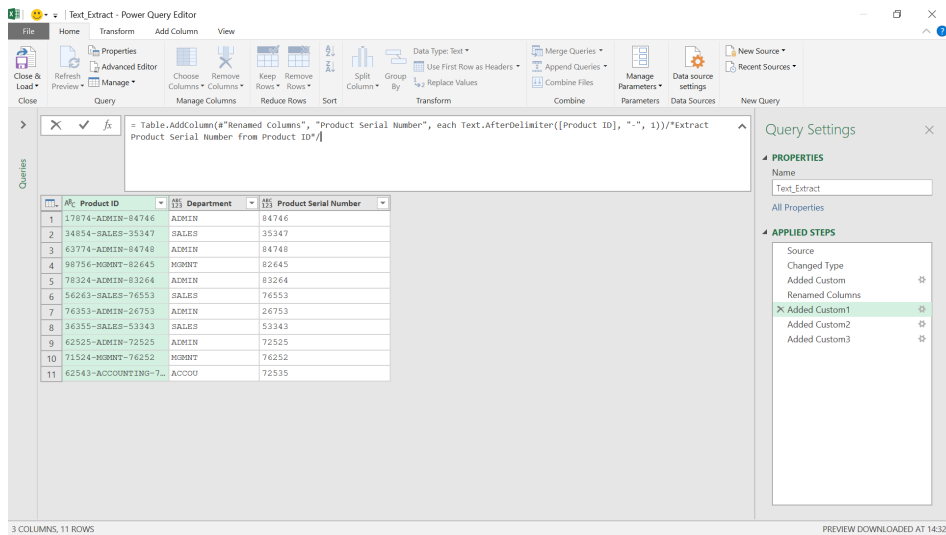
The comments can be added to long steps by using the arrow on the right of the Formula bar to expand the view.
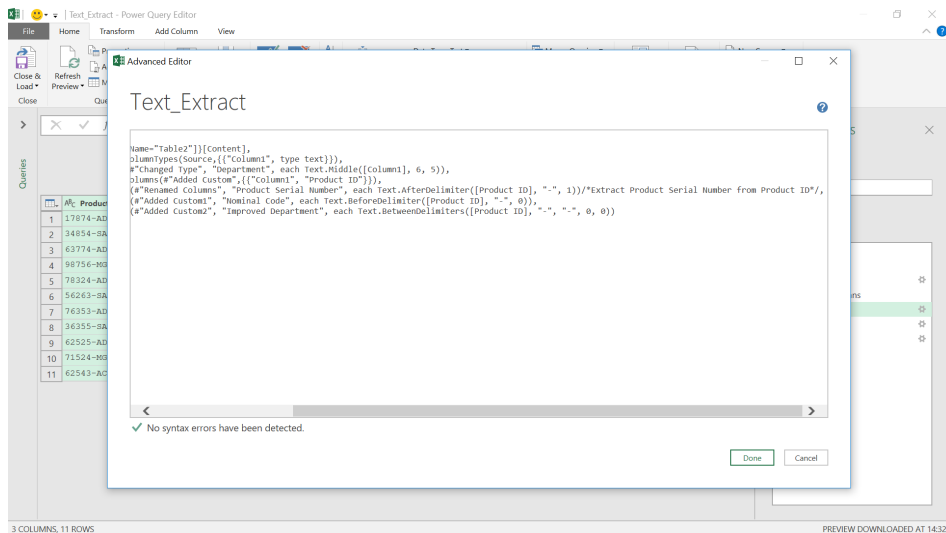
The comment is now visible in the Advanced Editor.
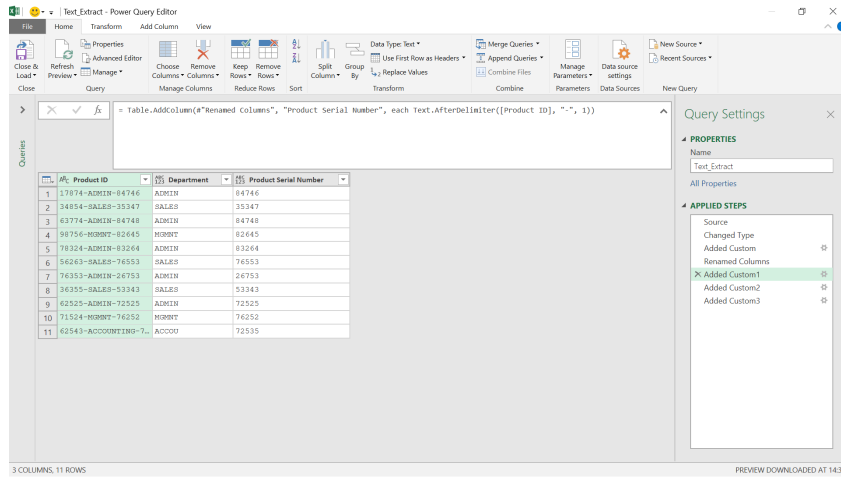
It is not however, visible from the Formula bar. See the next tip!

6. Avoiding the Advanced Editor when viewing comments

The sequel – how to avoid the Advanced Editor and view comments. The answer is to embed the comments in the formula.
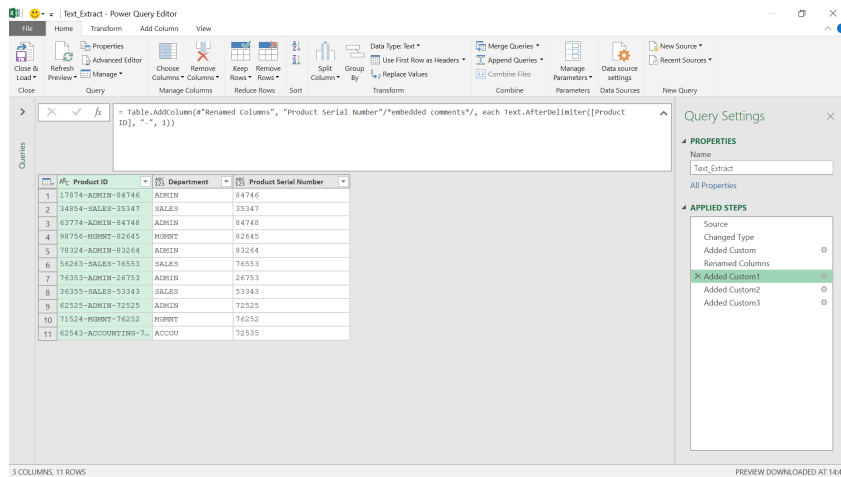
This time, if we exit the Formula bar and go back to it, we can see the comment. It's not the tidiest way to add comments, but it stops Power Query from stripping them out. I thought it might look neater after the '=' sign and before the main body of the formula, but if we put comments there they can only be seen in the Advanced Editor. At some point this method may not work, as Power Query seems to be programmed to prevent comments from appearing in the Formula Window, and this method is effectively a bug! For now, it seems to be the only way to view comments in the Formula Bar.

7. Change the properties of the query step so the purpose is clear

A typical list of steps generated by Power Query is not very descriptive.
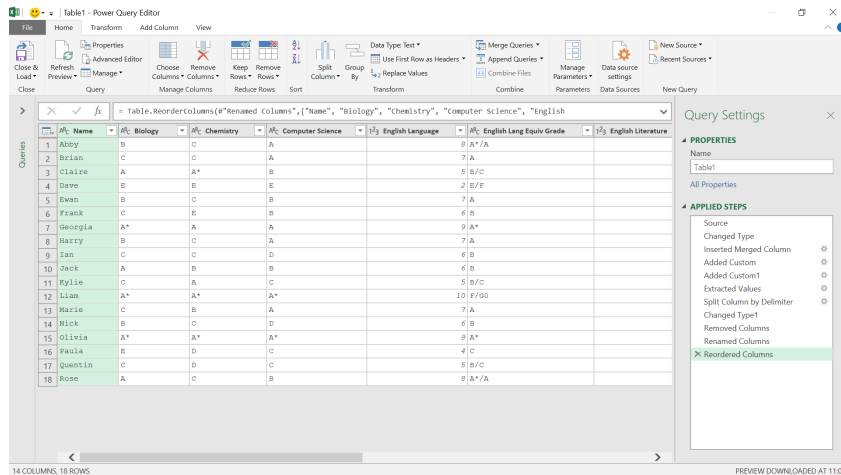
'Added Custom' and 'Added Custom1' are my particular favourites here. We can add comments in the Formula bar, as shown in the previous tip, but we can also edit the name and description of each step.
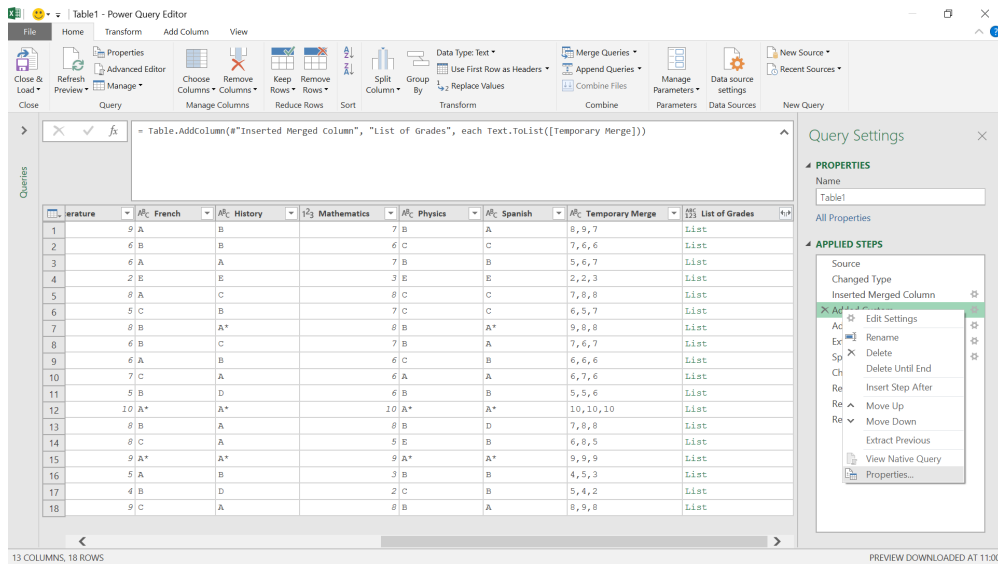
If we select a step and right-click, we have a 'Properties' option at the bottom:
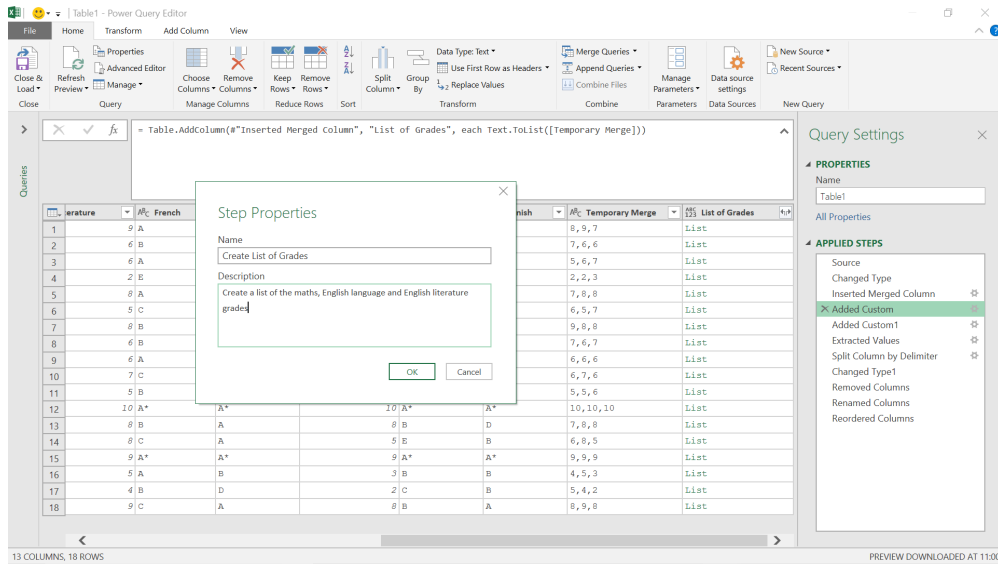
We can change the title and description to make what this step does clear.
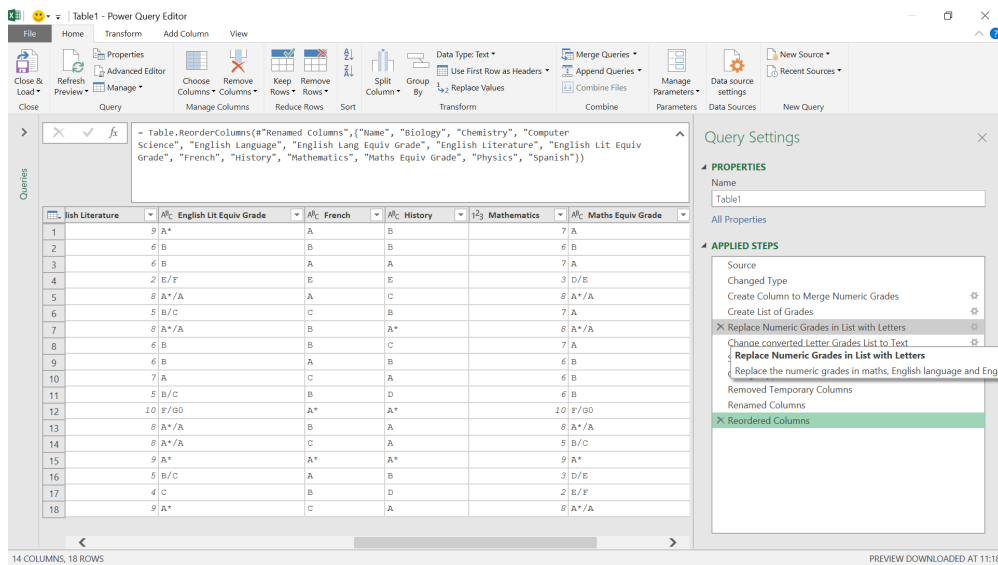
Once I have done this for several of the steps, we may not only see what is happening from the titles, but I can also hover over a step to see the description.

8. Add a row index to a query from the Excel worksheet

We can add an index from the Query Editor, but if we have a query associated with a table, then we can use another method.
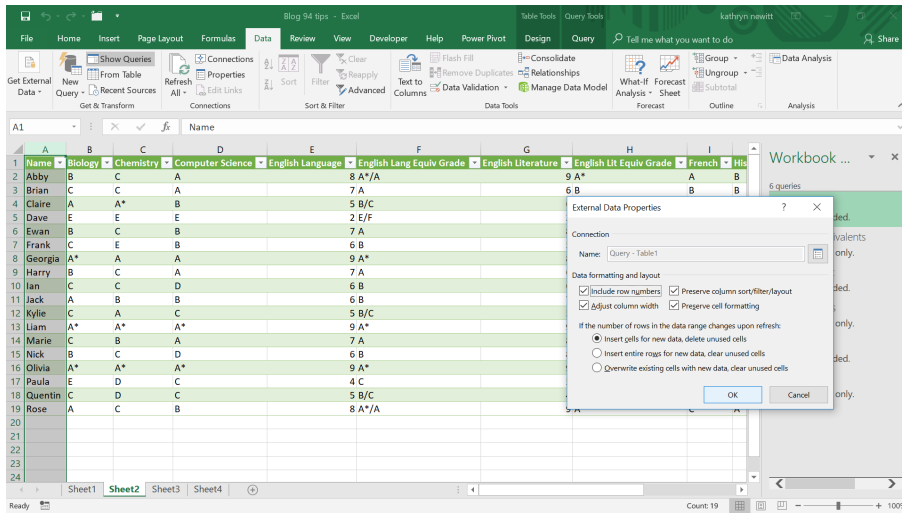
In the previous screenshot, we have selected the **Name** column, and in the 'Data' tab we have accessed 'Properties' from the 'Query' section. We have chosen to 'Include row numbers'. Let's click 'OK' and refresh 'Table1', which is the query associated with our data.
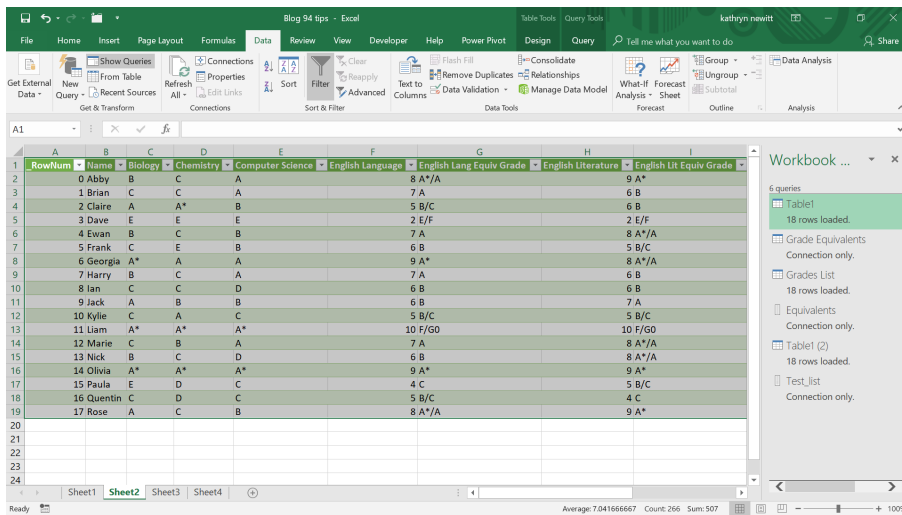
The column **RowNum** has been automatically added.

9. Split a query

Queries can get repetitive. Queries can get repetitive. If we want to create a new query which starts off in a very similar way to one I already have, then instead of copying it and deleting the parts we don't need, we can split a query.
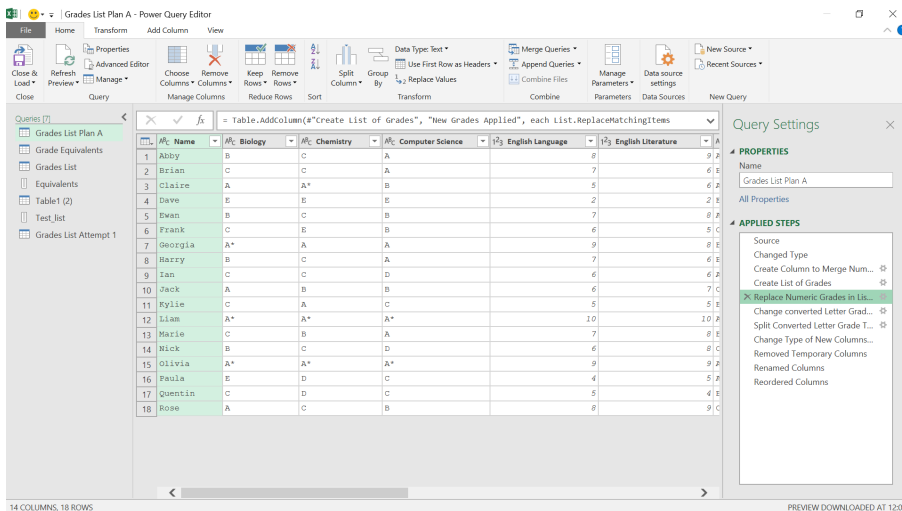
Here, we have decided to use a different approach to dealing with my grade scenario, and we want to compare the output. We will keep the 'Grades List Plan' query and create a new query 'Grades List First Step' which ends at the step 'Replace Numeric Grades in List':
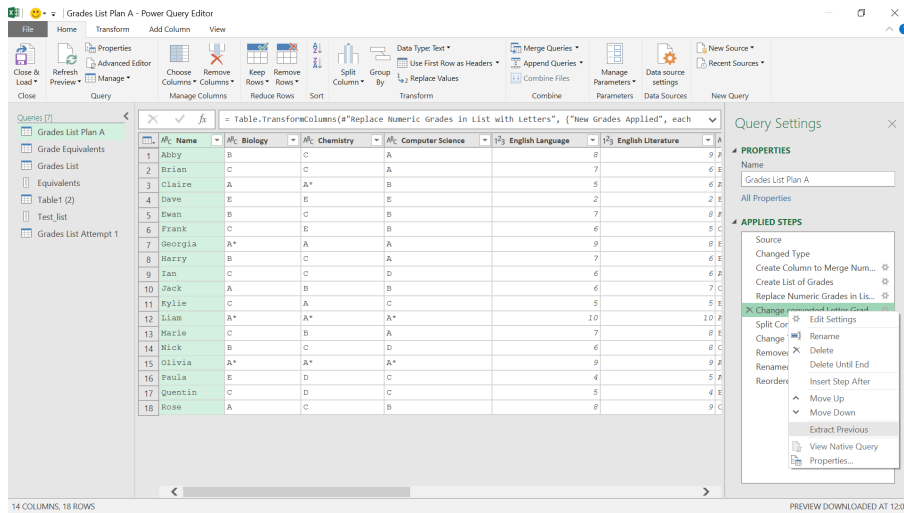
We move to the step after 'Replace Numeric Grades in List' and right-click. We have an option to 'Extract Previous'.
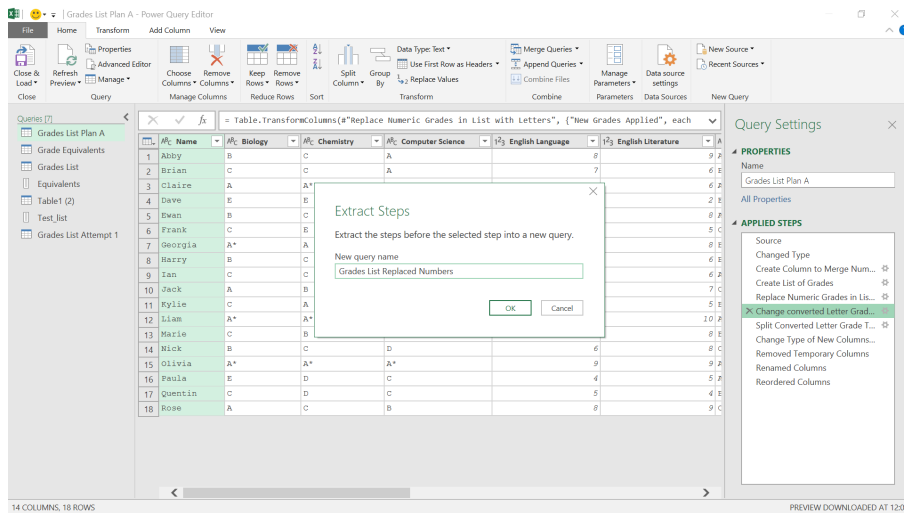
We'll choose the name for our new query and click 'OK'. Since this query will be the new source in 'Grade List Plan A', it's important that the name makes sense.
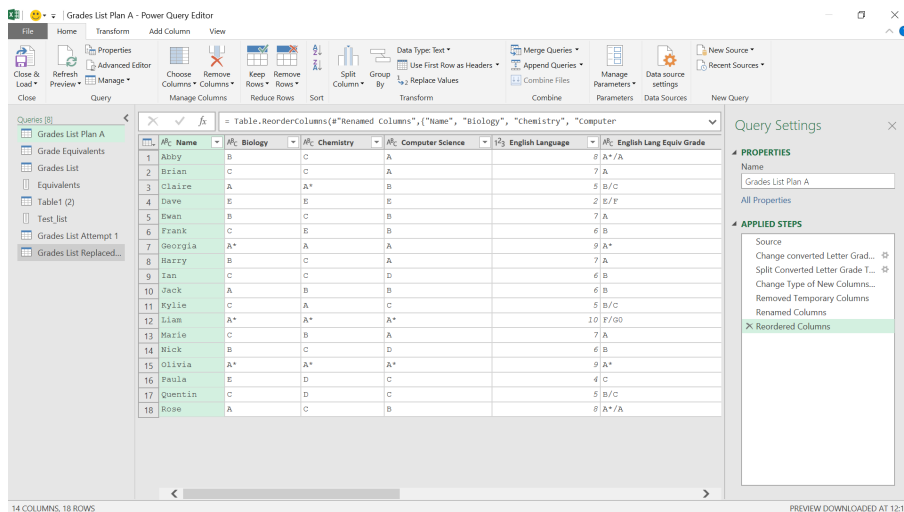
Power Query opens the query pane on the left where my new query appears. The steps in the new query have now disappeared from 'Grades List Plan A'.
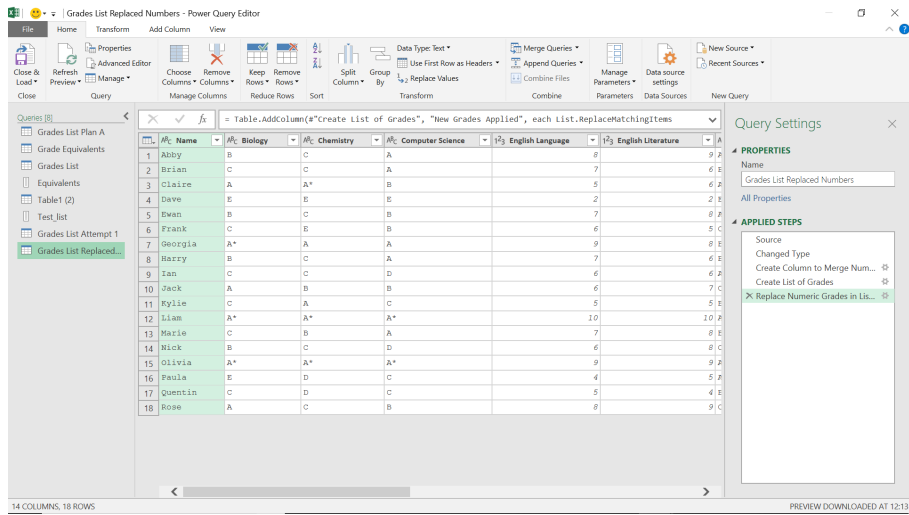
We can click on the query in the query pane to see that my new query has been created as we expected.
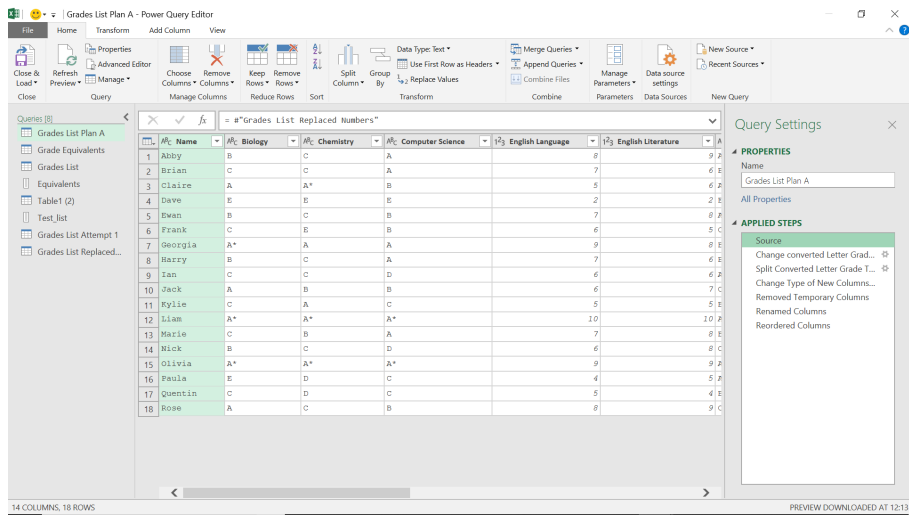
Our original query points to my new query as its source.
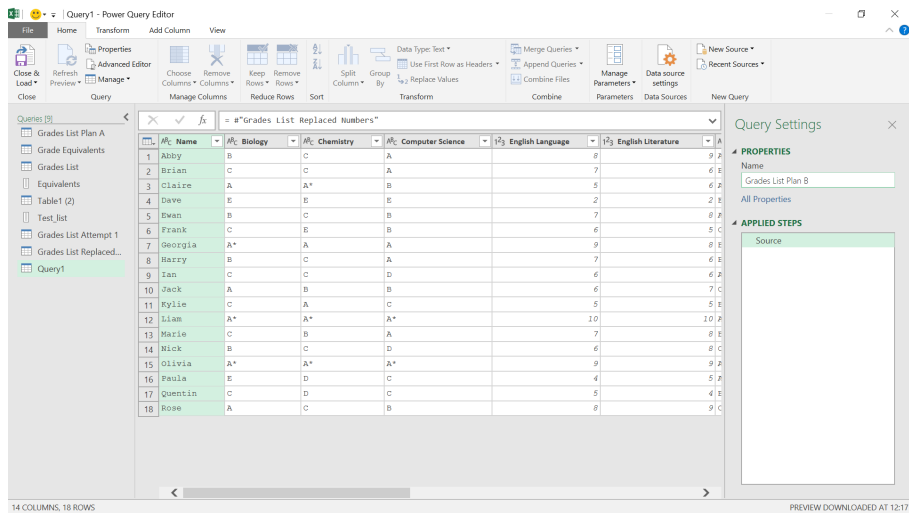
We can use the same source to start a query using our new ideas for the scenario. It's important that the source query is labelled correctly, as any changes to it will affect the dependent queries.

10. Keep learning

There are lots of articles on the **SumProduct** site about Power Query and other Excel tools. Microsoft frequently improves and changes functionality, and new ways to solve old problems appear not only in these articles, but throughout the Power Query community. One way to access useful information is to search for the *#powerquery* hashtag in Twitter.

Until next month.

Power BI Updates
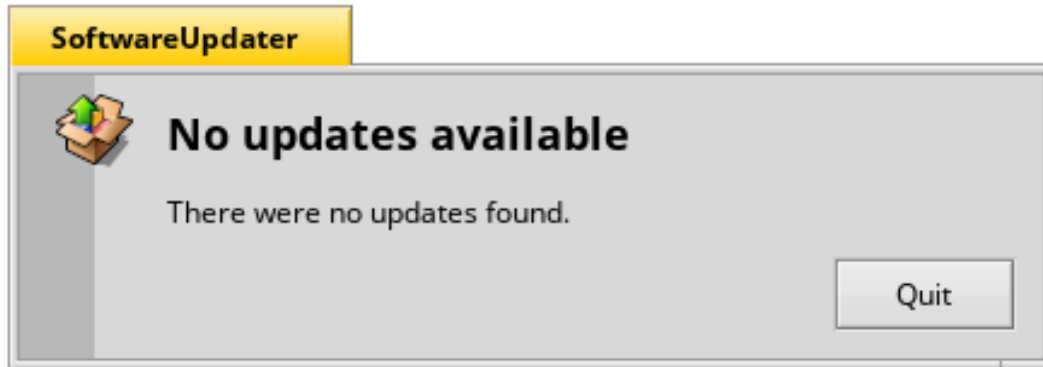
There will be updates for Power BI this month. Unfortunately, we've waited and waited and waited (honestly, it's like trying to get tickets for a Taylor Swift concert),
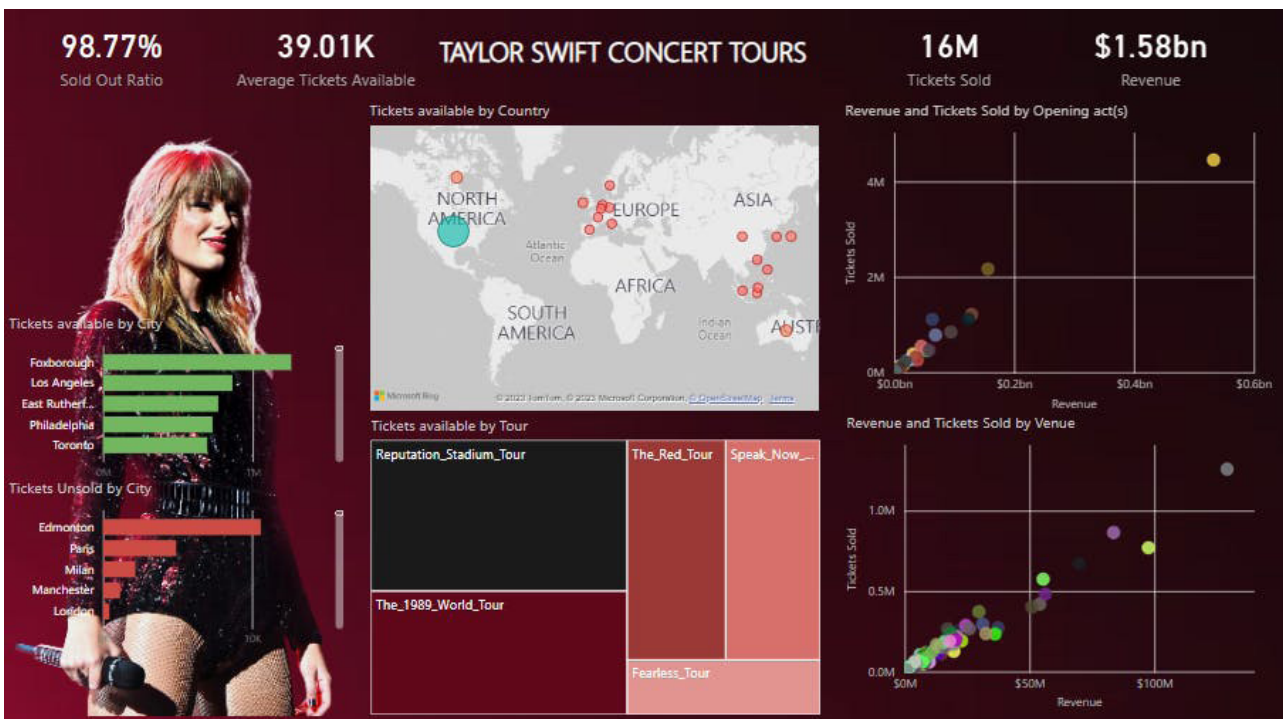
but our publishing deadline has beaten us for this newsletter. Looks like we will have to Shake It Off this month and put this topic into Exile. We know you know us All Too Well, but we couldn't resist leaving a Blank Space this month *[Stop! – Ed.]*

Until next month (hopefully!).

Connected Tables in Excel Online

Whilst there may be no Power BI Updates reviewed this month, there is a crossover item that fits nicely between Power BI and Excel newbies. Back in August last year, Microsoft announced new ways to create connected tables in Excel with Power BI. Since early March, this feature is now available to all in Excel online.

It goes further: Microsoft has also been working hard to ensure that all Microsoft 365 update channels are able to use this feature in the Excel desktop application.

To connect directly to data in your semantic model:

- Select 'Data from Power BI' on the Data tab of the Ribbon
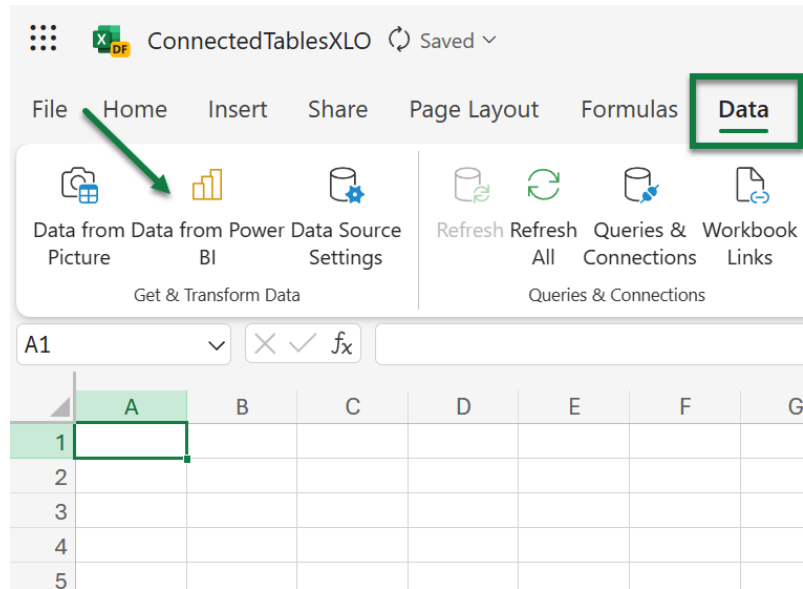
- Use the search feature in the 'Power BI Datasets' pane to find the dataset / semantic model you'd like to connect to and select 'Insert Table'
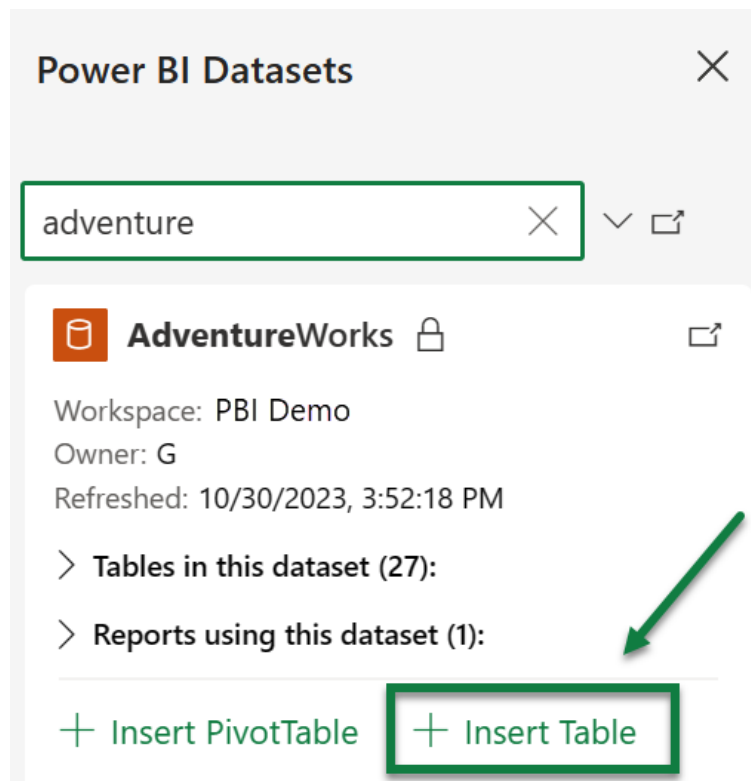
- Use the 'Create Table' dialog to:
  - select data to appear in your Table in the Data pane
  - apply any filters you'd like in the Filters pane
  - rearrange the field order in the Build pane
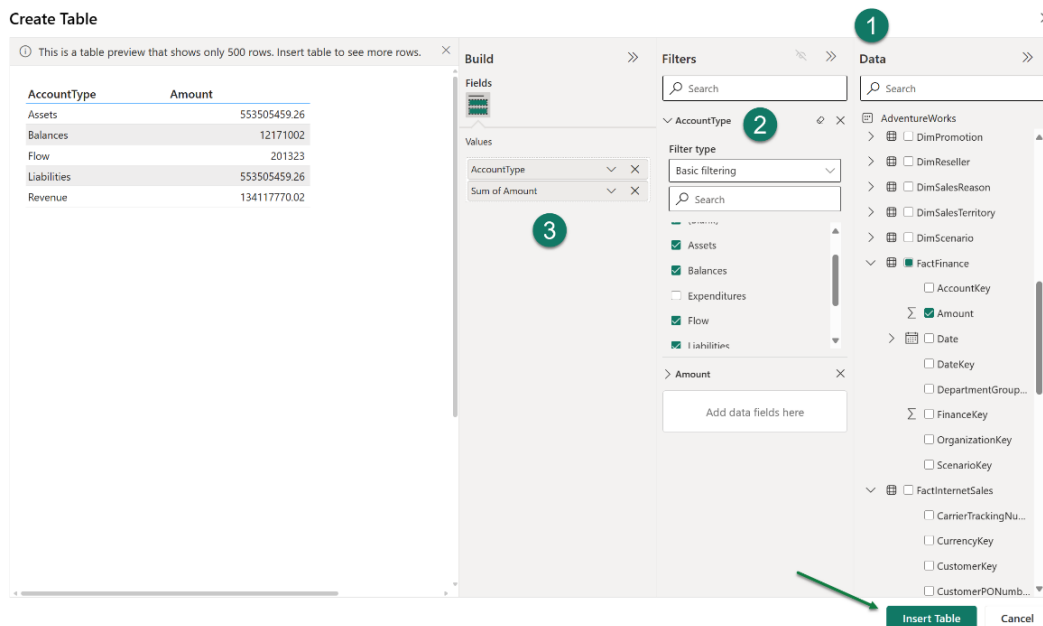
- Select 'Insert Table'. You now have a Table connected directly to your Power BI semantic model. Just like with other Connected Table experiences, the workbook contains a connection that you can review and modify in the Queries and Connections properties.

There are some limitations and other points to consider:

- After you select From Power BI in the Excel Ribbon, the artifacts loaded in the pane are not all the semantic models you have access to. They are a selection of your most used semantic models. You should use the search bar in the pane to get additional semantic models
- The order of Power BI semantic models in the pane may be different from their order in the Data hub in the Power BI Service
- For newly created semantic models, recently refreshed semantic models, or semantic models that you just got access to, it may take up to 24 hours for these semantic models to show up in the Power BI Semantic models pane. If you don't see the semantic model you want, navigate to the Power BI OneLake data hub (https://app.powerbi.com/datahub) from the Power BI Semantic models pane in Excel and use 'Analyze in Excel' to create an Excel workbook for that semantic model
- Excel and Power BI visuals use different query languages and data load behaviours, so data load performance may be different between Excel and Power BI.

There is an update on the availability of this feature for users of the Excel desktop application. The connected tables feature is currently available in the M365 Current Channel. It will be available in the Monthly Enterprise Channel in April 2024, and will be rolled out to the Semi-Annual Enterprise Channel in July 2024 as a part of the regular M365 update process.

New Features for Excel

This month sees the 'Chart data' task pane and the ability to use images and Data Types in your PivotTables both now available in Excel for the web. The new default theme for Office is Generally Available in Excel for Windows and Mac too.

You can check out the full list here:

Excel for the web

- • 'Chart data' task pane
- • Images and Data Types in PivotTables

Excel for Windows

- • New default theme for Office

Excel for Mac

- • New default theme for Office.

 Let's get started.

'Chart data' task pane

Already supported for Windows and Mac users, you can now use the 'Chart data' task pane to edit your chart's data on the web. The data task pane supports the following:

- • change a chart's source data range via range picker user interface [UI] (mouse selection)
- • expose new web-first UI to determine how data series are displayed on the chart (including which axis they are displayed on)
- • provide controls for configuring how hidden / empty cells appear and whether the source data is split by rows or columns.
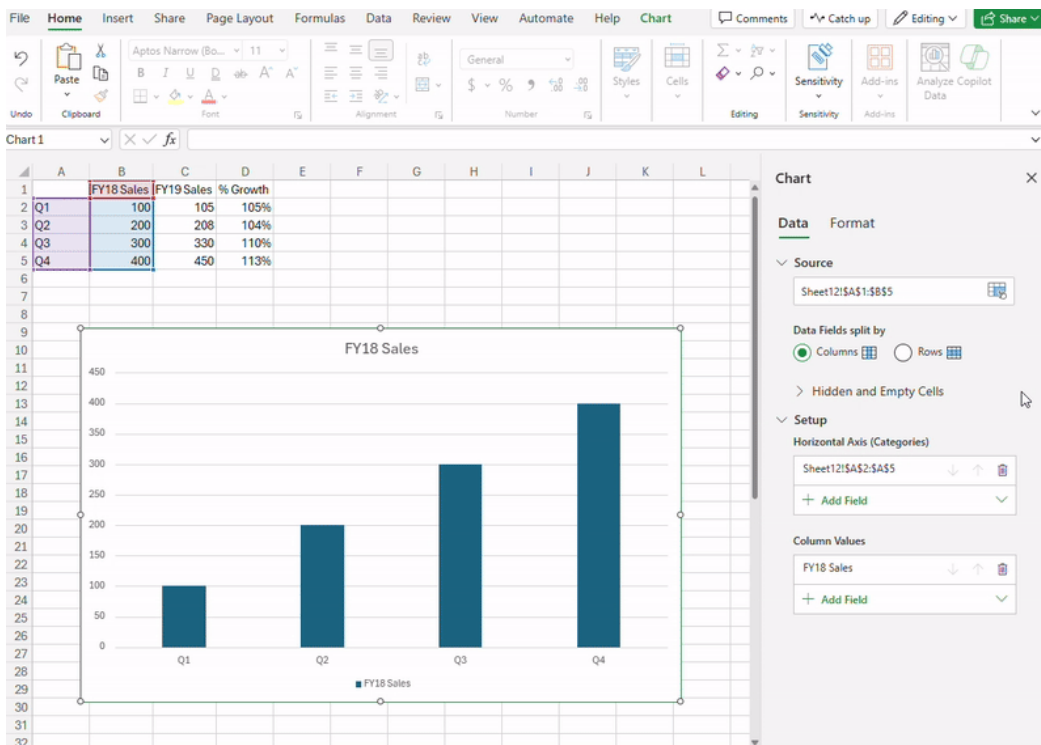
Images and Data Types in PivotTables

Images and Data Types in PivotTables are now available in Excel for the web. For example, we can use country flags in company with locations for a sales dataset. Let's consider a sample dataset from Microsoft:

We can change Data Type of the **Country** column to Geography (Excel for Windows depicted below):

Then, I can access other information from a Geography type (including flag images). We insert the above as a Table (**CTRL + T**) called **PivotData**. I insert a column **Flag** and use the dot operator (**.**) to get the flag images of the countries:

=[@Country].Image

Then we are able to insert a PivotTable and use images from the column **Flag**, in either row or column fields.

We can use inserted images as row and column fields. However, filters and slicers are not that sophisticated – yet!
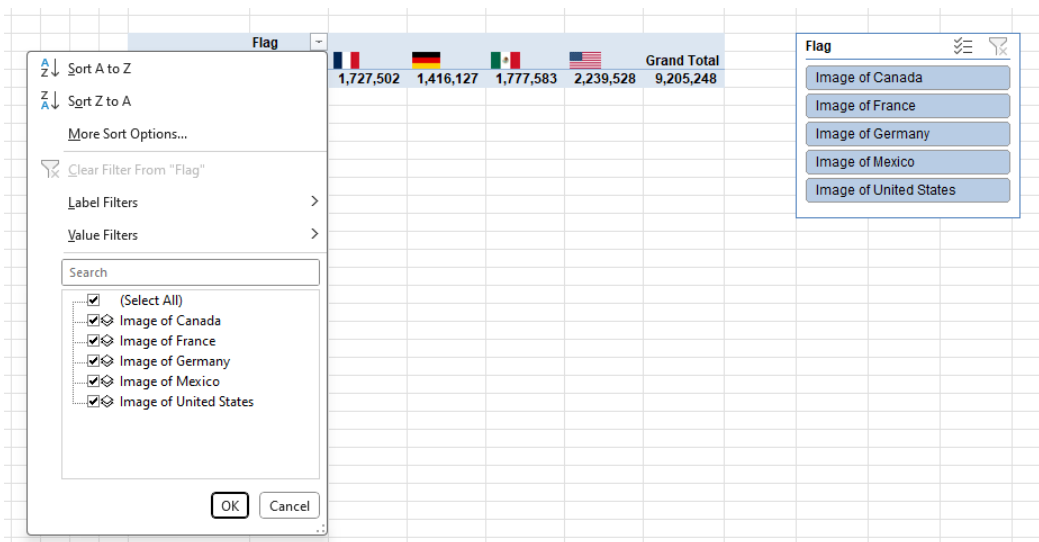
Do remember that this is a feature for pictures inserted into cells only; it is not for the classic floating images over cells. You should also note that presently we cannot load and process inserted images into Power Query:
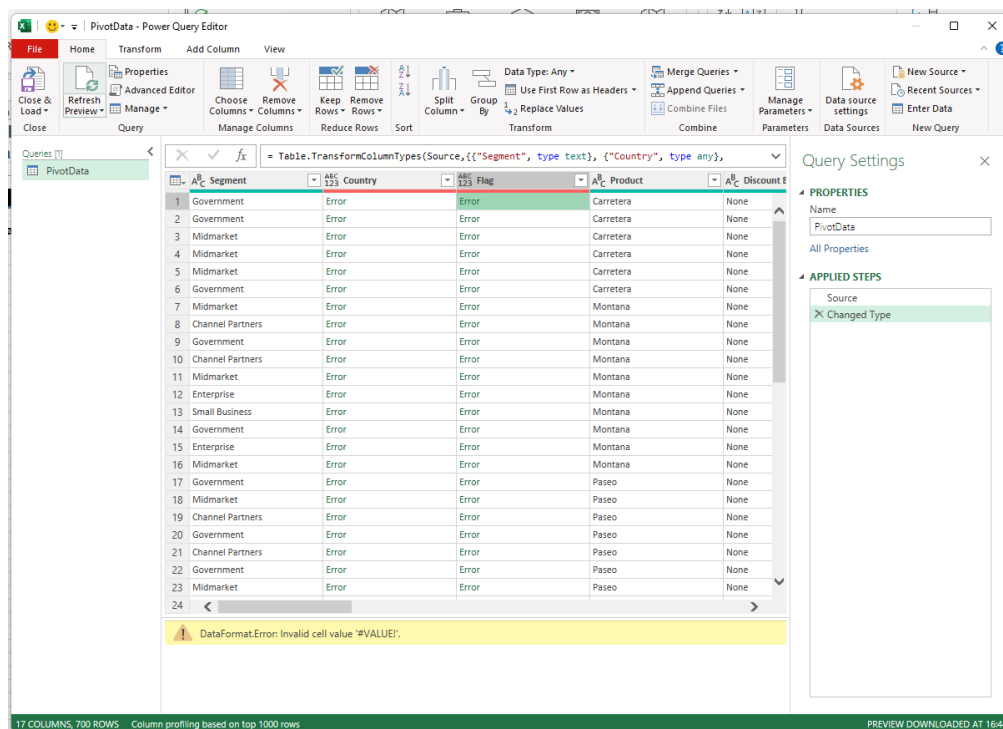
You can clearly see that both the **Flag** and the **Country** columns (Geography Data Type) are displaying *#VALUE!* errors.

New default theme for Office

Microsoft has refreshed the Office theme with a new default font, colour palette, style and line weights for Excel for Windows and Excel for Mac. I know life moves on, but I was happy with the old one.

Whilst deciding what to do, Microsoft released five] Calibri replacement candidate fonts to see what users thought:

1. Bierstadt
2. Grandview
3. Seaford
4. Skeena
5. Tenorite.

The popular choice was Bierstadt as the preferred font, and it has therefore been given a new name, **Aptos**. It supports a variety of weights and includes variants such as narrow, serif and monospace.
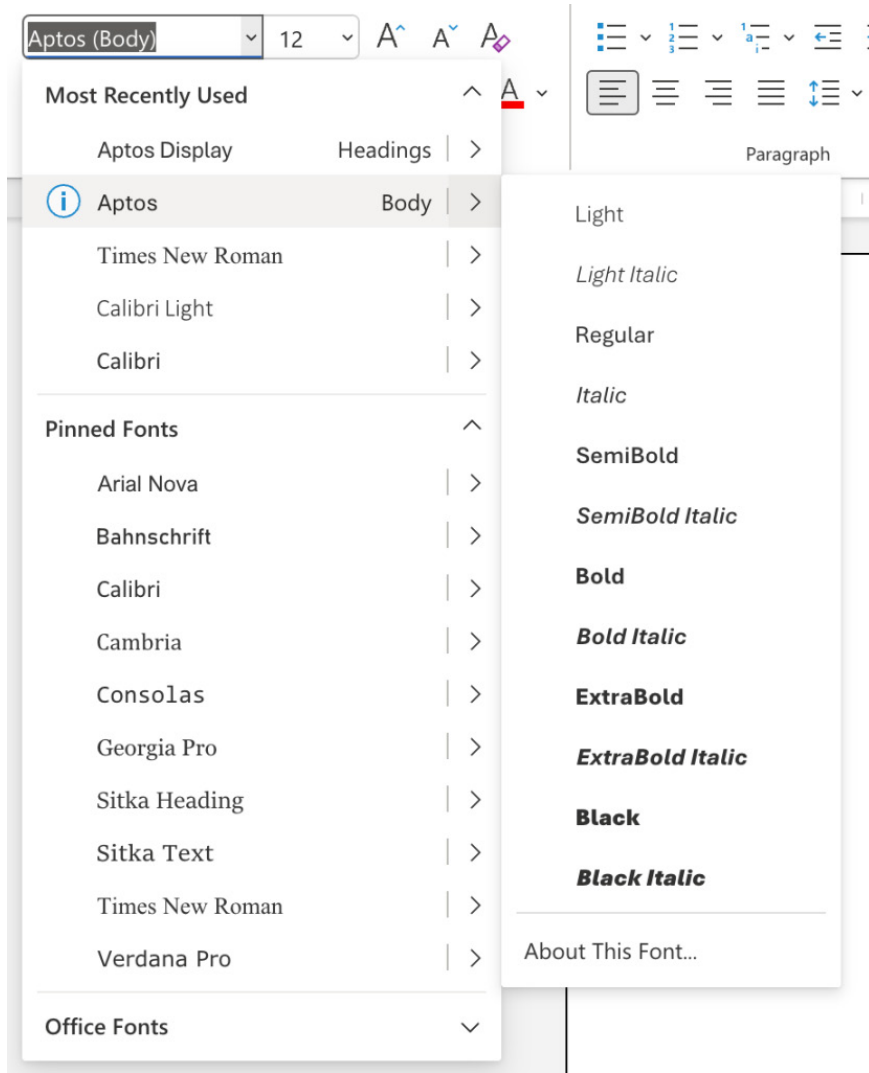
Microsoft also researched popular colour palettes and design trends, and created a set of default colours that works well in all Office applications.
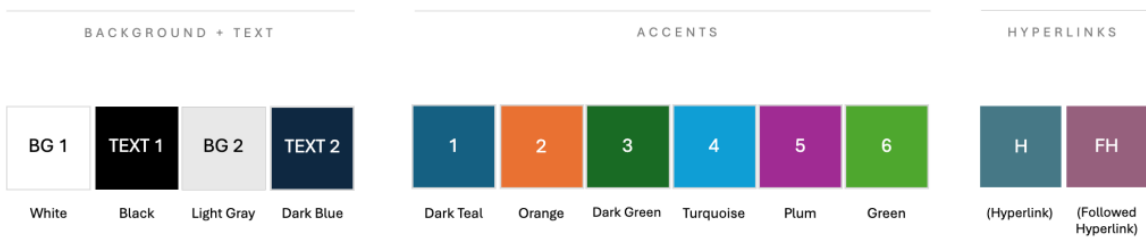
The default outline weights have been increased to improve the consistency between shapes and lines, as well as add better contrast.

The default style of Word documents and Outlook emails has been refreshed to make them easy to read, look more professional and easier to navigate.

Styles

AaBbCc Normal
AaBbCc No Spacing
AaBb... Heading 1
AaBbCc Heading 2
Aa... Title
AaBbCc Subtitle

AaBbCc Emphasis
AaBbCc Strong
AaBbCc Quote
AaBbCc Subtle Reference
AaBbCc Intense Reference
AaBbCc Book Title

AaBbCc List Paragraph
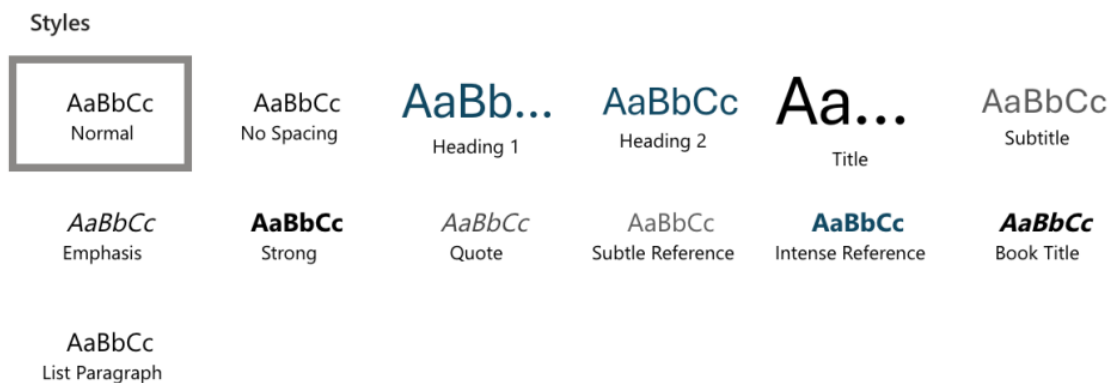
As the theme rolls out, all new documents, presentations, worksheets and emails you create in Word, PowerPoint, Excel and Outlook will use the new theme. You can decide for yourself what you think.

It should be noted that:

• your existing documents won't change. They will continue to use the original theme that was applied when they were created

  if you want to continue using the previous Office theme it's still there. It's been renamed Office Theme 2013 – 2022

• if neither Aptos nor Calibri are the right fonts for you, there are hundreds of other choices built into Microsoft 365. Simply choose the right typeface for you

• If you wish to revert to older style formatting, you can select 'Word 2013 formatting' from the Document Formatting.

This feature is available to Beta Channel users who are running:

• Win32: Version 2307 (Build 16626.20076) or later
• Mac: Version 16.75 (Build 23071400) or later
• Android: Version 16.0.16626.20076 or later
• Coming later to iOS.

You can check if a specific feature is in your version of Excel here.

Back next month, we're sure.

The A to Z of Excel Functions: MONTH

For some working in financial modelling, it appears that **MONTH** may be the closest they get to a date *(groan – Ed.)*. Seriously, working with time series in Excel is an important part of financial modelling. Since dates in Excel are merely formatted serial numbers counting from 1 January 1900 (usually), being able to identify the month from a date is not as trivial as you may first think.

Therefore, the **MONTH** function returns the month of a date by a serial number, from 1 (January) to 12 (December). It employs the following syntax to operate:

MONTH(serial_number)

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

• **serial_number**: this is required and represents the date of the month you are trying to find. Dates should be entered by using the **DATE** function or as the results of other formulae or functions. For example, you should use **DATE(2008,8,17)** for the 17th August 2008. Problems may occur if dates are entered as text.

It should be further noted that:

- Microsoft Excel stores dates as sequential serial numbers so they can be used in calculations. By default, January 1, 1900 is serial number 1, and January 1, 2008 is serial number 39448 because it is 39,448 days after January 1, 1900

- values returned by the **YEAR**, **MONTH** and **DAY** functions will be Gregorian values, regardless of the display format for the supplied date value. For example, if the display format of the supplied date is Hijri, the returned values for the **YEAR**, **MONTH** and **DAY** functions will still be values associated with the equivalent Gregorian date.

Please see our example below:

There is an alternative to this approach, depending upon:

- what you require
- accept that the result will be returned as a text data type, even if it has the appearance of a number.

The **TEXT** function allows you to change the way a number appears by applying formatting to it with format codes. It's useful in situations where you want to display numbers in a more readable format or you want to combine numbers with text or symbols.

However, as mentioned above, the **TEXT** function will convert numbers to text, which may make it difficult to reference in later calculations. It's best to keep your original value in one cell, then use the **TEXT** function in another cell. Then, if you need to build other formulae, always reference the original value and not the **TEXT** function result.

Its syntax is as follows:

=TEXT(value_to_be_formatted, "format_code")

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

- **value_to_be_formatted:** this is required and represents the value to be reformatted
- **"format_code":** this is required and must be in quotation marks. This represents the custom number formatting required.

For months formats, the following customer number format codes are relevant:

Date Code	Description
m	Month as a number without leading zeros (1 to 12)
mm	Month as a number with leading zeros (01 to 12)
mmm	Month as an abbreviation (Jan – Dec)
mmmm	Unabbreviated month (January – December)
mmmmm	First letter of month (J, F, M, A, M, J, J, A, S, O, N, D)

The A to Z of Excel Functions: MROUND

The **MROUND** function rounds a number to a desired multiple. There isn't much more to say about it, really (sorry). It employs the following syntax to operate:

MROUND(number, multiple)

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

- **number**: this is required and represents the value to be rounded
- **multiple**: also required, this is the **multiple** for which you wish to round the **number**.

It should be further noted that:

- **MROUND** rounds up, away from zero *(see image, below),* if the remainder of dividing the **number** by the given **multiple** is greater than or equal to half the value of the **multiple**
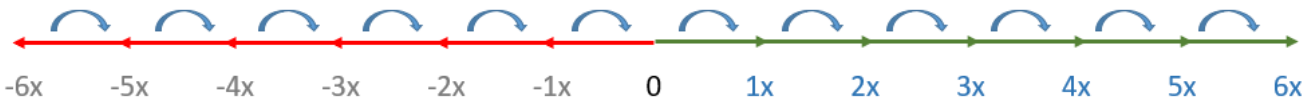
- the **number** and **multiple** arguments must have the same sign; if not, an *#NUM!* error is returned.

Please see our example below:

	A	B	C
1	Formula	Description	Result
2	=MROUND(10,3)	Rounds 10 to the nearest multiple of 3	9
3	=MROUND(-10,-3)	Rounds -10 to the nearest multiple of -3	-9
4	=MROUND(10.5,3)	Rounds 10.5 to the nearest multiple of 3	12
5	=MROUND(-10.5,-3)	Rounds -10.5 to the nearest multiple of -3	-12
6	=MROUND(-10.5,3)	Returns the #NUM! error as -10.5 and 3 have different signs.	#NUM!
7			

The A to Z of Excel Functions: MULTINOMIAL

$$(x_1 + x_2 + ... + x_k)^n = \sum \binom{n}{n_1 \, n_2 \, ... \, n_k} x_1^{n_1} \, x_2^{n_2} \, ... \, x_k^{n_k}$$

The multinomial distribution is a generalisation of the binomial distribution to two or more events.

In probability theory and statistics, the binomial distribution with parameters **n** and **p** is the discrete probability distribution of the number of successes in a sequence of **n** independent success / failure experiments, each of which yields success with probability **p**. For the record, a success / failure experiment is also called a Bernoulli experiment or Bernoulli trial. The binomial distribution is frequently used to model the number of successes in a sample of size **n** drawn with replacement from a population of size **N**.

Binomial Distribution
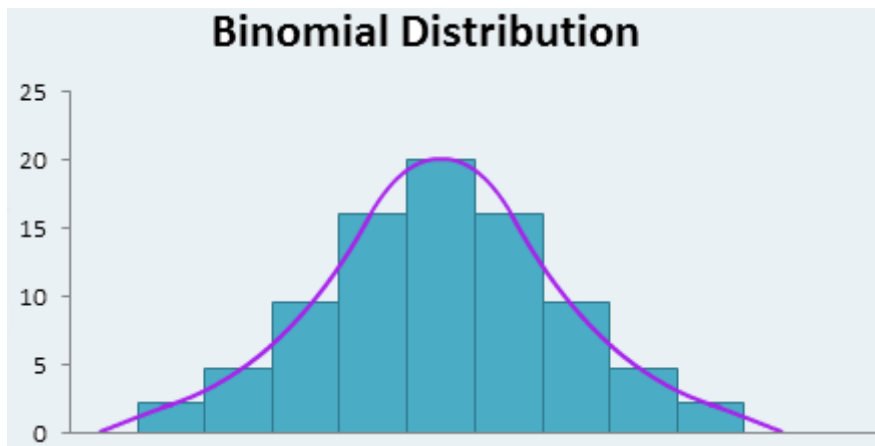
This function returns the individual term binomial distribution probability. The **BINOM.DIST** function should be used in problems with a fixed number of tests or trials, when the outcomes of any trial are only success or failure, when trials are independent and when the probability of success is constant throughout the experiment. For example, **BINOM.DIST** can calculate the probability that two of the next three babies born are male.

Therefore, for an experiment with the following characteristics:

- the experiment consists of **n** independent trials
- each trial has **k** mutually exclusive outcomes E_i
- for each trial the probability of outcome E_i is p_i,

let $x_1, ..., x_k$ be discrete random variables whose values are the number of times outcome E_i occurs in **n** trials. Then, $x_1, ..., x_k$ has a *multinomial* distribution. The (joint) probability distribution function (pdf) is defined as follows:

$$f(x_1, ..., x_k) = \frac{n!}{x_1! \cdot \, ... \, \cdot x_k!} p_1^{x_1} \cdot \, ... \, \cdot p_k^{x_k}$$

where

$$1 = \sum_{i=1}^{k} p_i$$

The case where k = 2 is equivalent to the binomial distribution.

Key properties of the multinomial distribution are:

- $E[x_i] = np_i$
- $var(x_i) = np_i(1-p_i)$
- $cov(x_i, x_j) = -np_i p_j$ for $i \neq j$.

The **MULTINOMIAL** function in Excel returns the ratio of the factorial of a sum of values to the product of factorials. It employs the following syntax to operate:

$$\text{MULTINOMIAL(number1, [number2, …])}$$

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

- **number1, number2, …**: **number1** is required (the rest are optional), and represents one [1] to 255 values for which you require the multinomial.

It should be noted that:

- if any argument is nonnumeric, **MULTINOMIAL** returns the *#VALUE!* error value
- if any argument is less than zero, **MULTINOMIAL** returns the *#NUM!* error value
- the multinomial is:

$$\text{MULTINOMIAL}(a_1, a_2, \ldots, a_n) = \frac{(a_1 + a_2 + \ldots + a_n)!}{a_1! \, a_2! \ldots a_n!}$$

As an example, suppose that a bag contains 12 balls: five [5] red, four [4] yellow and three [3] blue (5 + 4 + 3 = 12). You reach in the bag and pull out a ball at random and then replace it before making a subsequent selection. This experiment is repeated a total of 10 times. What is the probability that the outcome will result in exactly six [6] reds, two [2] yellows and two [2] blues (6 + 2 + 2 = 10)?

The possible outcomes for each trial in this experiment are E_1 = a red ball is drawn, E_2 = a yellow ball is drawn and E_3 = a blue ball is drawn. Thus p_1 = 5/12, p_2 = 4/12, p_3 = 3/12, x_1 = 6, x_2 = 2 and x_3 = 2, and so

$$f(6,2,2) = \frac{10!}{6! \, 2! \, 2!} \left(\frac{5}{12}\right)^6 \left(\frac{4}{12}\right)^2 \left(\frac{3}{12}\right)^2 = 0.045787$$

This has been reproduced in Excel using the **MULTINOMIAL** function (using three alternative approaches) below:

	A	B	C	D
1	Description	Variable	Assumption	Formula
2	No. of red balls	n_{red}	5	
3	No. of yellow balls	n_{yellow}	4	
4	No. of blue balls	n_{blue}	3	
5	Probability of a red ball taken	p_{red}	0.417	=C2/SUM($C$2:$C$4)
6	Probability of a yellow ball taken	p_{yellow}	0.333	=C3/SUM($C$2:$C$4)
7	Probability of a blue ball taken	p_{blue}	0.250	=C4/SUM($C$2:$C$4)
8	No. of red balls selected	x_{red}	6	
9	No. of yellow balls selected	x_{yellow}	2	
10	No. of blue balls selected	x_{blue}	2	
11	Probability of red balls	$p_{red}{}^{\wedge}x_{red}$	0.005233	=C5^C8
12	Probability of yellow balls	$p_{yellow}{}^{\wedge}x_{yellow}$	0.111111	=C6^C9
13	Probability of blue balls	$p_{blue}{}^{\wedge}x_{blue}$	0.062500	=C7^C10
14	Multinomial	Coeffecient	1260	=MULTINOMIAL(C8:C10)
15	Probability	Probability	0.045787	=C14*PRODUCT(C11:C13)
16		Alternative 1	0.045787	=PRODUCT(MULTINOMIAL(C8:C10),C5:C7^C8:C10)
17		Alternative 2	0.045787	=MULTINOMIAL(C8:C10)*EXP(SUMPRODUCT(C8:C10,LN(C5:C7)))
18				

The A to Z of Excel Functions: MUNIT

$$I_1 = [1], \quad I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad I_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \ldots, \quad I_n = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}.$$

In matrix algebra, the **MUNIT** function returns the unit matrix for the specified dimension. A unit matrix is simply a square (**n x n**) matrix which has the value of one [1] down the leading diagonal (from top left to bottom right) and zero [0] everywhere else.

Described like this, it doesn't sound very exciting!

It has the following syntax:

$$MUNIT(dimension)$$

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

- **dimension:** this is required and represents an integer specifying the dimension of the unit matrix that you wish to return, resulting in a square (**n x n**) array. This **dimension** must be greater than zero [0].

It uses the following equation:

$$1_{N \times N} = \begin{matrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{matrix}$$

If **dimension** is a value less than or equal to zero [0], **MUNIT** returns the *#VALUE!* error value.

If you are using Microsoft 365, then you can simply enter the formula in the top left-cell of the output range and then press **ENTER** to confirm the formula as a dynamic array formula. Otherwise, the formula must be entered as a legacy array formula by first selecting the output range, entering the formula in the top-left-cell of the output range, and then pressing **CTRL + SHIFT + ENTER** to confirm it. Excel inserts curly brackets (known as **braces**) at the beginning and end of the formula for you.

In Excel 365:
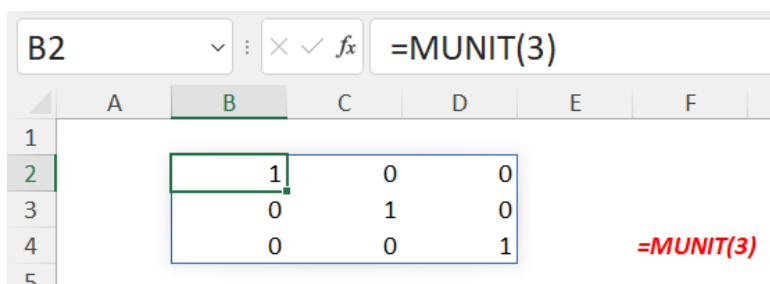
In a "legacy" version of Excel:
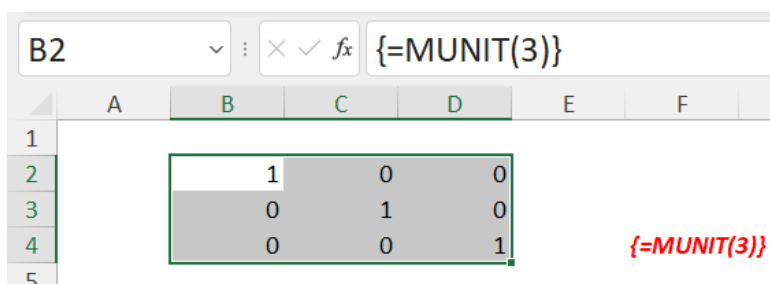
So what would you use this function for?

Let's keep this simple, and just consider a 2 x 2 matrix scenario. Matrices are used to transform data, often used in computer graphics, cartoons, *etc.* Sometimes, you wish to maintain a character's direction or bringing it nearer or closer (scaling up or scaling down).

In linear algebra, the term **eigenvector** is used to denote the direction that should remain consistent. The corresponding term **eigenvalue** would denote the scale of the enlargement, *e.g.*

- 1 would mean "no change"
- 2 would mean "double the length"
- -1 would mean point in the opposite direction to the eigenvalue.

Of course, mathematicians (such as myself!) have to complicate this notion in case someone still understands what I am talking about. It may be represented mathematically in the form

$$Av = \lambda v$$

where:

- **A** is a matrix of dimension **n** x **n**
- **v** is a vector, consisting of one row or one column, with **n** elements. This is the eigenvector
- **λ** is the scalar multiple (*i.e.* it is simply a number) which represents the eigenvalue.

[For what it's worth, "eigen" comes from the German word for "typical", if that helps clarify.]

As an example, let's consider the 2 x 2 matrix

$$\begin{pmatrix} -5 & 2 \\ 5 & 4 \end{pmatrix}$$

This has an eigenvector of $\begin{pmatrix} 2 \\ 10 \end{pmatrix}$ with an associated eigenvalue of five [5].

How on earth was this calculated? Let's revert to the original premiss:

Av = λv

To ensure I am considering like with like, I will make sure both sides of the equation are matrices, so I will add the 2 x 2 identity matrix, *i.e.* **MUNIT(2)** here:

Av = λIv

where **I** is the unitary 2 x 2 matrix. Thus:

Av − λIv = 0

Assuming the eigenvector **v** is non-zero, we may solve for the eigenvalue **λ** by calculating the determinant of the matrix. This is a special calculation in linear algebra, denoted by **|A|**, meaning the determinant of the square matrix **A**. For a 2 x 2 matrix, the determinant is calculated as

$$\left| \begin{pmatrix} a & b \\ c & d \end{pmatrix} \right| = \text{ad} - \text{bc}$$

To solve for **λ**, we calculate

|A − λI| = 0

Therefore,

$$\left| \begin{pmatrix} -5 & 2 \\ 5 & 4 \end{pmatrix} - \lambda \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right| = 0$$

which reduces to

$$\left| \begin{matrix} -5 - \lambda & 2 \\ 5 & 4 - \lambda \end{matrix} \right| = 0$$

Calculating this determinant, we get

(-5 − λ)(4 − λ) − (2 x 5) = 0

Which simplifies to a quadratic equation, *viz.*

λ² + λ − 30 = 0

which is

(λ + 6)(λ − 5) = 0

i.e. **λ** equals either -6 or +5. That means there are <u>two</u> distinct eigenvalues, one of which is five [5]. Knowing this, we may now calculate the associated eigenvector:

$$\begin{pmatrix} -5 & 2 \\ 5 & 4 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 5 \begin{pmatrix} x \\ y \end{pmatrix}$$

Using standard matrix algebra, we get the two equations

-5x + 2y = 5x

5x + 4y = 5y

Bringing all values over to the left-hand side of the equations, we get

-10x + 2y = 0

5x − y = 0

Clearly, **y = 5x**, so an eigenvector is any non-zero multiple of the vector $\begin{pmatrix} 1 \\ 5 \end{pmatrix}$. For example,

$$\begin{pmatrix} -5 & 2 \\ 5 & 4 \end{pmatrix} \begin{pmatrix} 2 \\ 10 \end{pmatrix} = \begin{pmatrix} 10 \\ 50 \end{pmatrix} = 5 \begin{pmatrix} 2 \\ 10 \end{pmatrix}$$

i.e. **Av = λv**

If you wanted to calculate this in Excel, you would need **MUNIT**. Further, if you add up the digits on the leading diagonal (known as the **trace** of the matrix), you will note

$$-5 + 4 = -1$$

The two solutions for **λ** are -6 and +5, which also add up to -1. This is because the sum of the eigenvalues of a matrix equals the trace of that matrix. This can be a good check, which **MUNIT** may also be used for:

trace(A) =SUM(A * MUNIT(ROWS(A)))

So ends today's mathematics lesson. Class dismissed.

More Excel Functions next month.

Beat the Boredom Suggested Solution

Earlier in this month's newsletter, we posted an interesting problem, whereby Power Pivot would return with an error when we loaded a Table with two very similar columns in them.
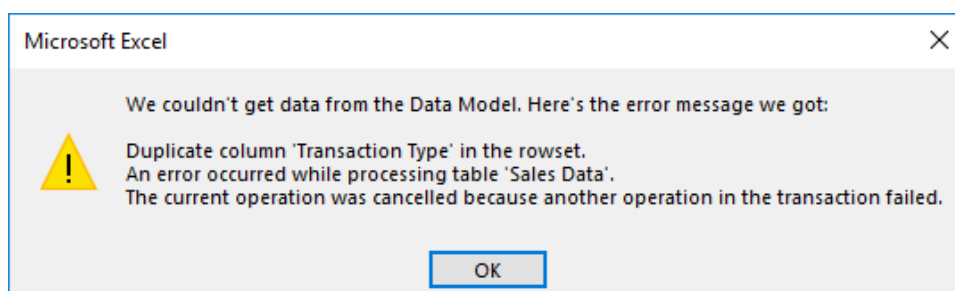
Do note that these columns were accepted by Power Query and even successfully exported into a Table in Excel.

So, how do we get around this?

The answer may surprise you. Change the column names! If you had not noticed, the original table had two very similar column names 'Transaction Type' and 'Transaction type':

Customer ID	Customer Name	Date	Country	Region	Transaction Type	Internal / External	Amount
1004	Harmonic Sonics	5/09/2018	AU	BBE	Payments	Internal	711
1001	Bizuplyz	29/10/2019	AU	SPA	Payments	External	4,459
1003	L. R. Repeat	25/12/2017	AU	JUY	Payments	Internal	644
1003	L. R. Repeat	20/11/2018	AU	SPA	Payments	Internal	2,472
1003	L. R. Repeat	8/03/2019	AU	JUY	Payments	Internal	5,917
1004	Harmonic Sonics	13/08/2018	AU	SPA	Payments	External	2,352
1003	L. R. Repeat	8/05/2020	AU	GFY	Payments	Internal	508
1003	L. R. Repeat	12/05/2020	AU	SPA	Payments	External	3,749
1004	Harmonic Sonics	29/06/2018	AU	GFY	Payments	External	2,764
1002	Plumb'n'Stuff	22/02/2019	AU	SPA	Payments	Internal	675
1002	Plumb'n'Stuff	7/06/2019	AU	GFY	Payments	External	5,716
1004	Harmonic Sonics	23/01/2019	AU	JUY	Payments	External	756
1002	Plumb'n'Stuff	2/12/2018	AU	NHJ	Payments	Internal	2,543
1004	Harmonic Sonics	10/07/2018	AU	GFY	Payments	External	5,516
1002	Plumb'n'Stuff	17/03/2020	AU	NHJ	Payments	External	3,989

The two columns were only distinguished by the capitalised 'T' in 'type'. Power Pivot does not recognise the capitalisation as a distinguishing factor, therefore treated both columns as duplicates. Therefore, the solution is to simply change the column name:

Customer ID	Customer Name	Date	Country	Region	Transaction Type	Internal / External	Amount
1004	Harmonic Sonics	5/09/2018	AU	BBE	Payments	Internal	711
1001	Bizuplyz	29/10/2019	AU	SPA	Payments	External	4,459
1003	L. R. Repeat	25/12/2017	AU	JUY	Payments	Internal	644
1003	L. R. Repeat	20/11/2018	AU	SPA	Payments	Internal	2,472
1003	L. R. Repeat	8/03/2019	AU	JUY	Payments	Internal	5,917
1004	Harmonic Sonics	13/08/2018	AU	SPA	Payments	External	2,352
1003	L. R. Repeat	8/05/2020	AU	GFY	Payments	Internal	508
1003	L. R. Repeat	12/05/2020	AU	SPA	Payments	External	3,749
1004	Harmonic Sonics	29/06/2018	AU	GFY	Payments	External	2,764
1002	Plumb'n'Stuff	22/02/2019	AU	SPA	Payments	Internal	675
1002	Plumb'n'Stuff	7/06/2019	AU	GFY	Payments	External	5,716
1004	Harmonic Sonics	23/01/2019	AU	JUY	Payments	External	756
1002	Plumb'n'Stuff	2/12/2018	AU	NHJ	Payments	Internal	2,543
1004	Harmonic Sonics	10/07/2018	AU	GFY	Payments	External	5,516
1002	Plumb'n'Stuff	17/03/2020	AU	NHJ	Payments	External	3,989

The column 'Transaction type' has been renamed to 'Internal / External'. Let's try to load this Table into our Data Model:
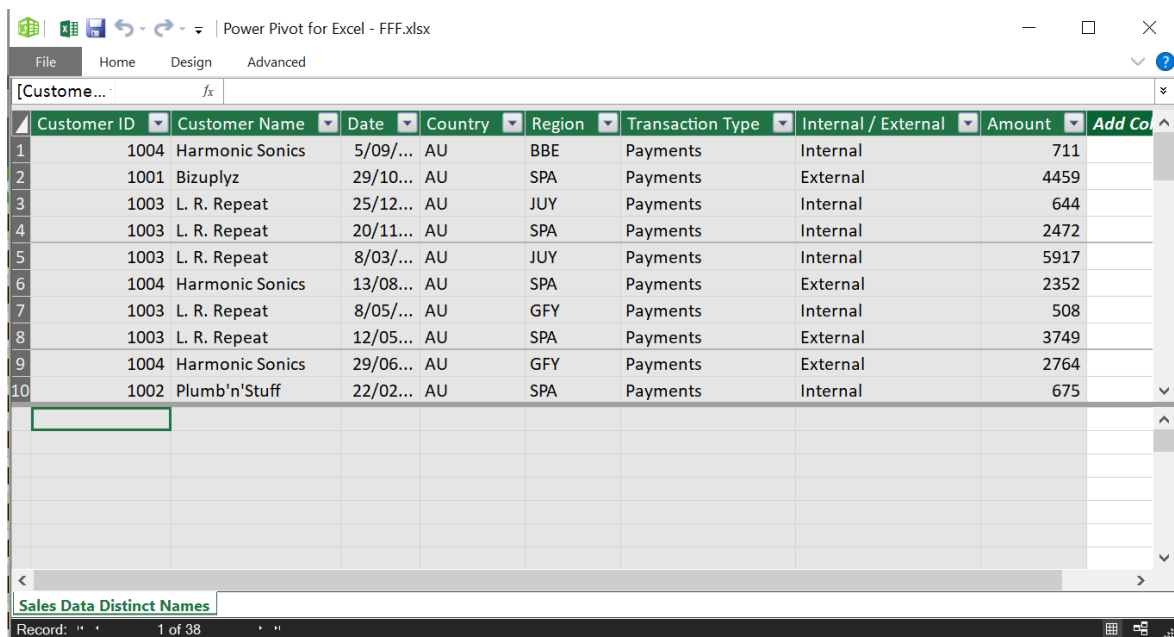
Success! We are now able to work with this data table in Power Pivot. So how did you go? Did you find a solution other than changing the column names? Let us know, we'd be keen to hear if you think you have a better way of circumventing this error. Otherwise, more next month.

Upcoming SumProduct Training Courses

Location	Course	Course Date	Local TIme	UTC	Duration
Virtual (Australia)	ChatGPT Part 2	8 April	09:00 - 12:30 AEST	7 April, 23:00 UTC - 8 April, 02:30 UTC	1 Day

Key Strokes

Each newsletter, we'd like to introduce you to useful keystrokes you may or may not be aware of. We've started going through the alphabet actions. We see the light this month – albeit the **UV** light…

Keystroke	What it does
CTRL + U	Underline (toggle)
CTRL + SHIFT + U	Expand / collapse Formula bar
CTRL + ALT + SHIFT + U	Adds French accent aigu
ALT + V	Excel 2003 and prior: activate View menu
CTRL + V	Paste
CTRL + ALT + V	'Paste Special' dialog

There are *c.*550 keyboard shortcuts in Excel. For a comprehensive list, please download our Excel file at http://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.