SumProduct

NEWSLETTER #149 - April 2025
www.sumproduct.com | www.sumproduct.com/thought

Are you a spreadsheet champion?

This month, we highlight the Next Generation of modellers. Quite a star trek in the making, we think. We'll stop the puns there though as we're in danger of orbiting Uranus. This month, we bring *Spreadsheet Champions* to your attention as "a documentary about the biggest competition you've never heard of" is publicised.

We're also proud to mention we've been recognised as on of the 60 Best Excel Blogs by FeedSpot, so we'd better live up to that and work on our newsletter next.

Talking of which, it's a little smaller this month with no Power BI Updates to present (there is an MVP Summit happening presently to be fair which has probably got Microsoft preoccupied), but there is still plenty to keep you interested. There is the usual Beat the Boredom Challenge, Charts & Dashboards Tips, Over to AI, Excel for Mac, Visual Basics, Power Pivot Principles, Power Query Pointers, Excel Updates, plus we look at the permutations available for the functions in the A to Z of Excel functions and continue to play Twister with the Keyboard Shortcuts to wrap up.

Happy reading and remember: stay safe, stay happy, stay healthy.

Liam Bastick, Managing Director, SumProduct

Spreadsheet Champions

It's always good to talk about the next generation of modellers. I must confess this event has passed us by – but let's rectify that right now. Hosted for over 20 years, the Certiport MOS World Championship is an annual competition that brings together the best, brightest and apparently youngest from all over the world to showcase their proficiency in Microsoft Office applications. Of course, of all the categories, the Excel competition stands tallest as a test of analytical and problem-solving skills. Participants are tasked with complex challenges that require not only technical know-how but also creativity and precision.

The documentary *Spreadsheet Champions* from GoodThing Productions captures the excitement and intensity of the MOS World Championship, offering viewers a glimpse into the lives of the competitors and the challenges they face. The documentary not only celebrates their achievements but also highlights the broader significance of Excel skills in today's world.

And who happens to be one of the talking heads in this celebration of all things youth and Excel? Our very own **Liam Bastick** had to stick his sticky beak in there somewhere. What does he know about youth..?

Billed as a "a documentary about the biggest competition you've never heard of", you can find out more here: https://www.spreadsheetchampions.com/.

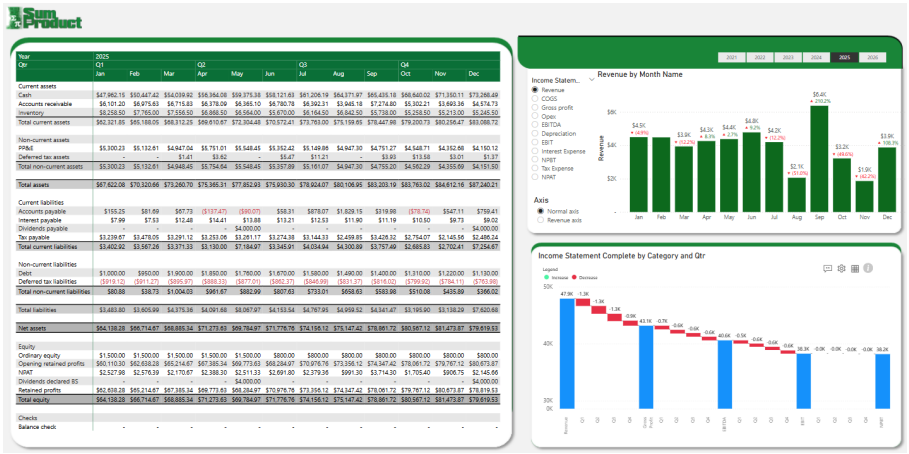
Top 60 Excel Blogs

With 1.2 million hits on our website in the past year, SumProduct is proud to announce it has been recognised as one of the "60 Best Excel Blogs" for 2025 by FeedSpot (https://bloggers.feedspot.com/excel_blogs/). With a truly astounding 72 Twitter followers (what's Twitter?), we guess the blogs are based on other metrics and not our anti-social media policy!

Coming in at Number 9 on the list, it's always nice to be recognised in the industry. We'll keep publishing nearly 30 articles a month so do make sure you keep up to date with our ever-growing and ever-more-recognised website.

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…*



In Power BI, there is a visualisation called the Matrix visualisation (no Keanu Reeves though). We can use it to display numerical values over several time periods. For examples, for 2021 *(say)*:

Year	2021									
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct
Total Sales	1,000.00	1,100.00	1,150.00	950.00	900.00	800.00	980.00	600.00	1,050.00	1,200.00
Total COGS	450.00	525.00	537.50	487.50	475.00	450.00	370.00	275.00	387.50	550.00
Gross Profit	550.00	575.00	612.50	462.50	425.00	350.00	610.00	325.00	662.50	650.00

Each line item is a measure. The challenge here is to make the **Gross Profit** measure stand out more by inserting lines into the Matrix visualisation like so:

Year	2021									
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct
Total Sales	1,000.00	1,100.00	1,150.00	950.00	900.00	800.00	980.00	600.00	1,050.00	1,200.00
Total COGS	450.00	525.00	537.50	487.50	475.00	450.00	370.00	275.00	387.50	550.00
Gross Profit	550.00	575.00	612.50	462.50	425.00	350.00	610.00	325.00	662.50	650.00

No, we did not simply draw a line on top and below the **Gross Profit** measure. We can expand the visualisation by breaking the **Total COGS** measure down to **Direct Labour** and **Direct Materials**, by adding these in and *the lines move automatically*:

Year	2021									
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct
Total Sales	1,000.00	1,100.00	1,150.00	950.00	900.00	800.00	980.00	600.00	1,050.00	1,200.00
COGS										
Direct Labour	200.00	250.00	250.00	250.00	250.00	250.00	125.00	125.00	125.00	250.00
Direct Materials	250.00	275.00	287.50	237.50	225.00	200.00	245.00	150.00	262.50	300.00
Total COGS	450.00	525.00	537.50	487.50	475.00	450.00	370.00	275.00	387.50	550.00
Gross Profit	550.00	575.00	612.50	462.50	425.00	350.00	610.00	325.00	662.50	650.00

How did we do it? This can be achieved "easily" in Power BI, with no lines drawn whatsoever.

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

Over to AI

Here at SumProduct we write many articles for many magazines and our authoring contracts have been revised recently to ensure Artificial Intelligence hasn't taken over our duties. We can assure everyone they have not.

But it did get us thinking: *what if we did let AI loose and write a regular article for the newsletter?*

This month, we asked AI what it thought was the current flavour of the month in Excel.

This Month's Prompt: To future proof your business, which is the better way to build a forecast financial model. Should you use normal (legacy) Excel, dynamic arrays in Excel, Python in Excel or Power BI? Justify your answers with examples.

In today's ever-evolving business landscape, future-proofing your enterprise involves making informed decisions about the tools and technologies you use. One critical aspect is the creation of accurate and adaptable forecast financial models. This article explores four [4] popular tools:

1. normal (legacy) Excel
2. dynamic arrays in Excel
3. Python in Excel
4. Power BI

and provides insights into their effectiveness for building robust financial models.

1. NORMAL (LEGACY) EXCEL

Normal Excel has been the go-to tool for financial modelling for decades. Its widespread use and familiarity make it a convenient choice for many businesses. However, it has its limitations.

Advantages:

- **widespread familiarity:** most finance professionals are well-versed in Excel
- **ease of use:** Excel provides a user-friendly interface for basic financial modelling.

Disadvantages:

- **manual updates:** legacy Excel models often require significant manual intervention
- **error-prone:** high risk of human error due to manual data entry and formula management.

Consider a simple cash flow forecast model where revenues and expenses are manually entered and updated on a monthly basis. While this approach works, it becomes increasingly cumbersome and error-prone as the business grows.

DYNAMIC ARRAYS IN EXCEL

Dynamic arrays are a recent addition to Excel that offer significant improvements over legacy Excel.

Advantages:

- **automation:** dynamic arrays can automatically update calculations as data changes
- **reduced errors:** minimises human error by automating repetitive tasks
- **scalability:** better suited for handling larger datasets and more complex calculations.

Disadvantages:

- **learning curve:** users may need time to adapt to new functions and capabilities.

This could be used for a revenue forecast model that uses dynamic arrays to pull data from various sources and dynamically update calculations as new data is added can significantly reduce manual effort and increase accuracy.

PYTHON IN EXCEL

Integrating Python with Excel offers a powerful combination for advanced financial modelling.

Advantages:

- **advanced analytics:** Python provides robust libraries for statistical analysis and machine learning
- **automation:** Python scripts can automate complex and repetitive tasks
- **data integration:** easily integrates with various data sources for comprehensive analysis.

Disadvantages:

- **technical expertise:** requires knowledge of Python programming
- **complexity:** may be overkill for simple models.

One example could be a financial forecast model that uses Python to perform advanced scenario analysis, incorporating machine learning algorithms to predict future trends based on historical data, provides deeper insights and more accurate predictions.

Power BI

Power BI is a business analytics tool that offers advanced data visualisation and analysis capabilities.

Advantages:

- **real-time insights:** Power BI can connect to live data sources and provide real-time updates
- **data visualisation:** offers sophisticated visualisations to communicate insights effectively
- **integration:** seamlessly integrates with other Microsoft products and various data sources.

Disadvantages:

- **learning curve:** requires time to master its advanced features
- **cost:** may involve additional costs for premium features and licences.

An example could be a financial dashboard in Power BI that pulls in data from accounting software, CRM systems and market databases to provide a holistic view of the business's financial health, enabling dynamic and interactive forecasting.

CONCLUSION

Choosing the right tool for building a forecast financial model depends on your specific needs and resources. For basic models, normal Excel or dynamic arrays in Excel may suffice. For more advanced analytics, Python in Excel offers powerful capabilities, whilst Power BI provides comprehensive data integration and visualization. Each tool has its strengths and weaknesses and the best choice will depend upon factors such as the complexity of your models, the size of your datasets and the technical expertise available within your team.

By carefully considering these factors, you can select the most suitable tool to future-proof your business and ensure accurate financial forecasting.

What do you think? It would have been nice if it had explained each term and you feel if I had asked it to build a financial model using regular expressions it might have come up with an argument. A proper recommendation would have been nice, and perhaps the importance of the key difference between “Conclusion” and “Recommendation” in your prompt.

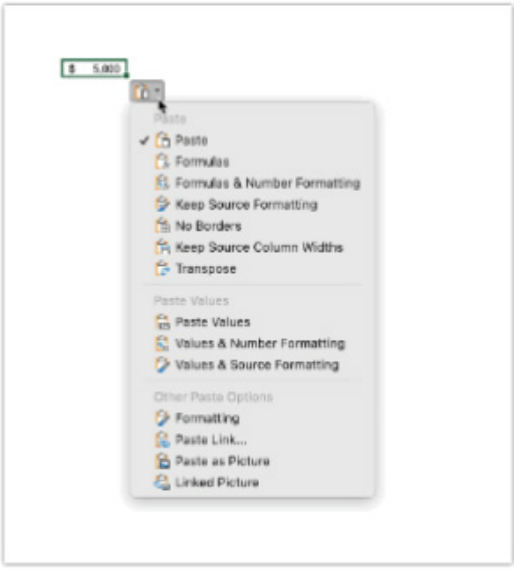
Do you agree AI will get there this year? It will be interesting to see precisely what happens. Send us your comments and ideas for questions to contact@sumproduct.com.

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, in our series about Microsoft Excel for Mac, we show you a few handy pop-up buttons that are unfortunately not going to pop up if you’re using Excel on a Mac.

At certain times, Excel will show you a pop-up button depending on the action you’ve taken or what you currently have selected. An example that’s often overlooked is what you see after you paste into a range. Excel shows a pop-up button called the ‘Paste Options’ button. The folks on Microsoft’s Excel engineering team often refer to it as the “paste recovery” menu. You may press it to get options regarding the paste action that you just performed.

In the example below, you can see a button appear after pasting a value into a cell. Expanding the button’s menu provides all the different paste actions that are available so you can change what was just pasted.



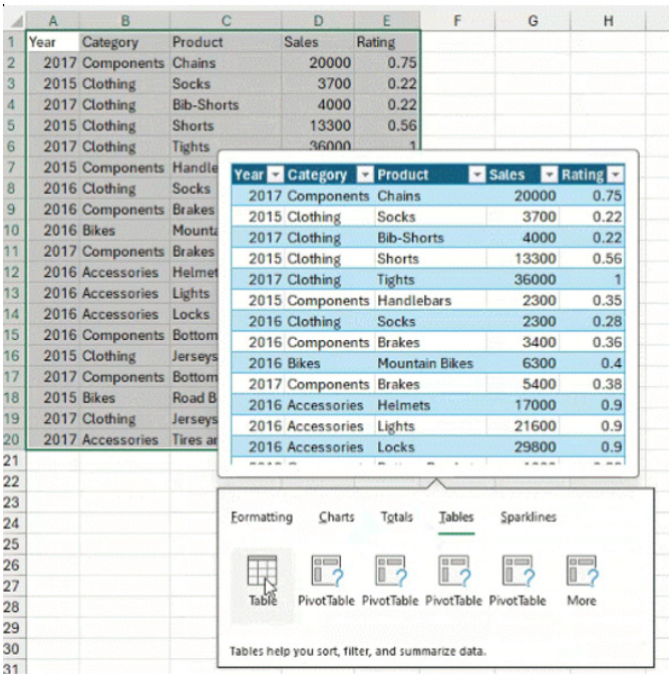
What's Missing on Mac?

Unfortunately, there are several similar pop-up buttons that don't show up if you're using Excel for Mac. The good news is that although the buttons are there to help you do things a bit easier, their absence on Mac doesn't mean you can't get the same result. You just won't be able to take the shortcuts offered by the buttons on Windows.

Here are the places where you won't find pop-up buttons on Mac:

- On the corner of charts to let you quickly add elements, change format, or add a filter

- At the edge of a range selection: the 'Quick Analysis' button that helps apply a variety of quick changes based upon the selection:



There may be others, but those are the ones that we're aware of and that you most frequently encounter on Windows.

Word to the Wise

As you can see from the options that are shown screenshots we've included, you can accomplish the same actions, you just need to do it a different way on Mac.

We'll continue 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 start a series on some of the common traps we sometimes fall for.

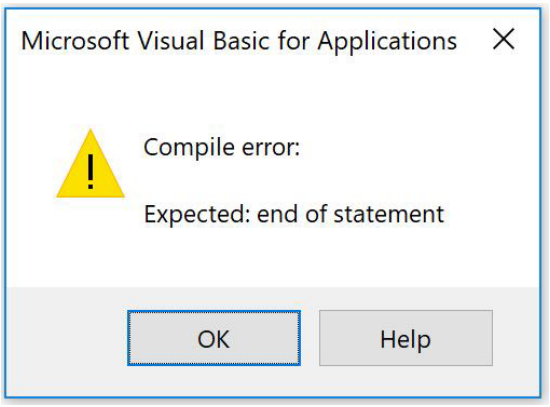
Suppose I'm entering in a simple formula that checks a value for a particular name and provides an answer. It might look something like this:

=IF(G2="Amy",1,IF(G2="Billy",2,3))

If I'm using VBA to populate this, I might think that I can just use the following line of code:

rng.Formula = "IF(G2="Amy",1,IF(G2="Billy",2,3))"

However, this quickly gives me an error:




```
Sub Put_Formulae_In_Cells()

Dim rng As Range

For Each rng In Range("E3:E5")

    rng.Formula = "=IF(G2="Amy",1,IF(G2="Billy",2,3))"

Next rng

End Sub
```

What's happened here? Using quotation marks has meant that the VBA code is processing the second set of quotation marks just before Amy as the end of the text string and is therefore expecting the code to stop at that point (or do something to add onto the text string, such as concatenate additional items).

What can we do to avoid this problem? Well, if we want to tell VBA that we want to use those quotes as part of the text string, we need to type it in twice:

rng.Formula = "IF(G2=""Amy"",1,IF(G2=""Billy"",2,3))"

This tells VBA that the quotation mark isn't the end of the string – it's merely a character that we want to use. Then, the formula works perfectly:

```
Sub Put_Formulae_In_Cells()

Dim rng As Range

For Each rng In Range("E3:E5")

    rng.Formula = "IF(G2=""Amy"",1,IF(G2=""Billy"",2,3))"

Next rng

End Sub
```

Hopefully you won't get caught out using hard-coded parameters from now on.

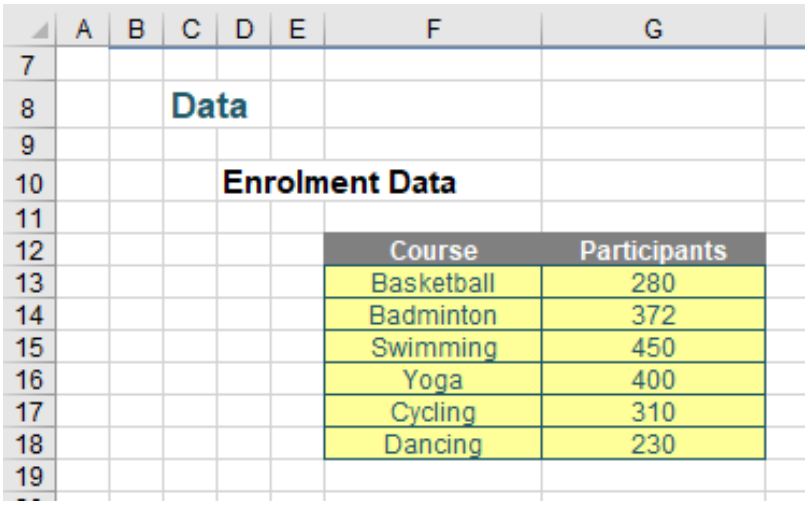
More next month…

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 creating a Bar chart with dynamic ordering.

Bar Charts are not only useful to illustrate the magnitude of values in one or more data series, but also a good alternative to the Pie Chart, which compares the proportion of a series in a group of data.

For example, here is some example enrolment data for six [6] sport courses from a youth centre.

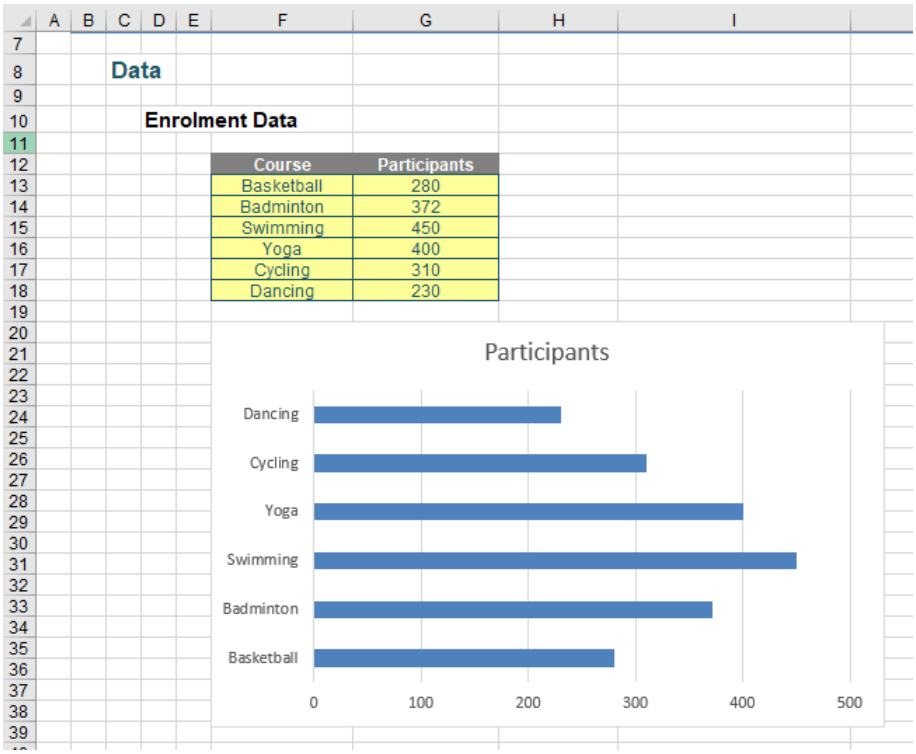


Data

Enrolment Data

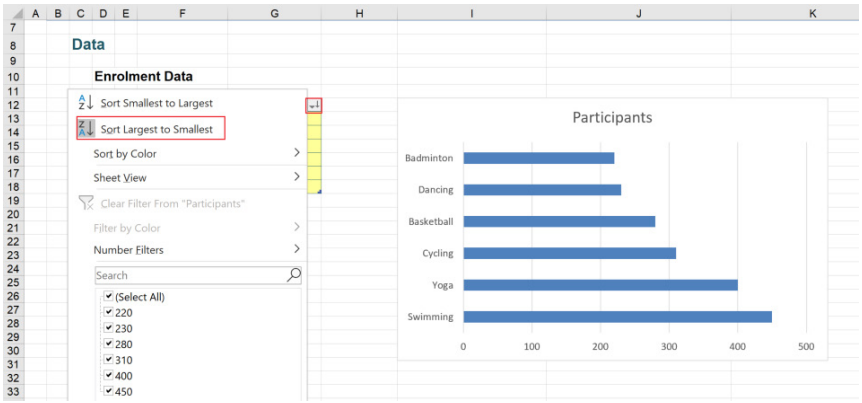
Course	Participants
Basketball	280
Badminton	372
Swimming	450
Yoga	400
Cycling	310
Dancing	230

From the above data, if we create a Bar chart, it may appear as below:



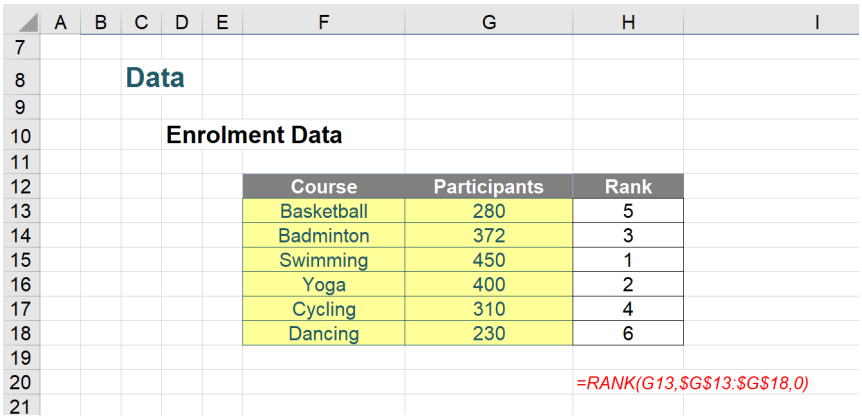
We want the Bar chart to illustrate the percentage of each course's participants relative to the total and also sort the bars in order from largest to smallest.

One way to sort the bars is to turn the data in cells **F12:G18** to a table by highlighting the range and press **CTRL + T**, then sort the Participants column in the descending order. However, when we update the data in the table, we need to sort the data again so that the chart works in the way we want.



Let's consider another way to get this working without using tables. In column **H**, we will get the order of the bar using the **RANK** function. The formula in cell **H13** is:

=RANK(G13,$G$13:$G$18,0)



Data

Enrolment Data

Course	Participants	Rank
Basketball	280	5
Badminton	372	3
Swimming	450	1
Yoga	400	2
Cycling	310	4
Dancing	230	6

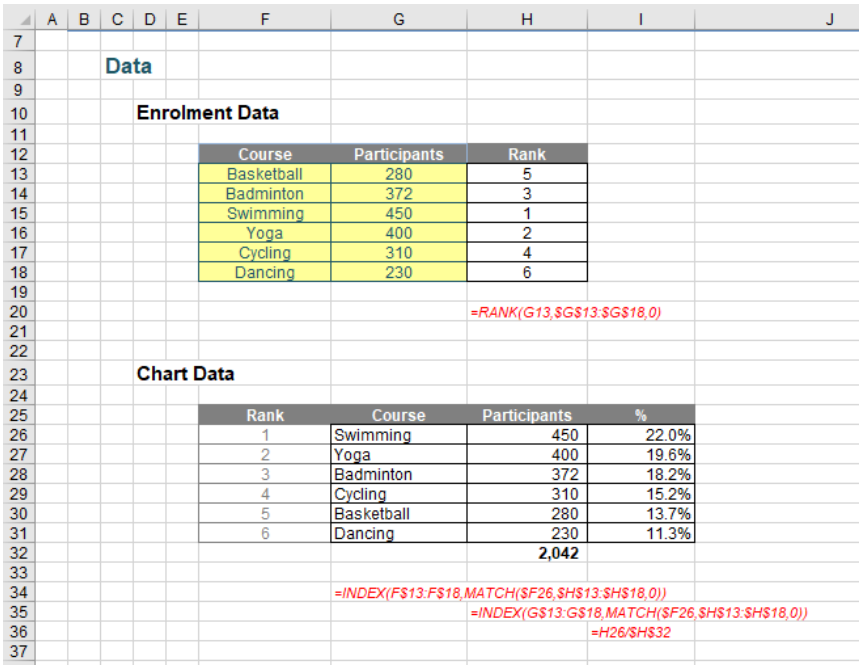
=RANK(G13,$G$13:$G$18,0)

Next, we will create a **Chart Data** section to get the related data in order. We will use the common formulaic combination **INDEX MATCH** against the **Rank** to get the **Course** and **Participants**. The formulae in cells **G26**, **H26** and **I26** are respectively:

=INDEX(F$13:F$18,MATCH($F26,$H$13:$H$18,0))

=INDEX(G$13:G$18,MATCH($F26,$H$13:$H$18,0))

=H26/$H$32



Data

Enrolment Data

Course	Participants	Rank
Basketball	280	5
Badminton	372	3
Swimming	450	1
Yoga	400	2
Cycling	310	4
Dancing	230	6

=RANK(G13,$G$13:$G$18,0)

Chart Data

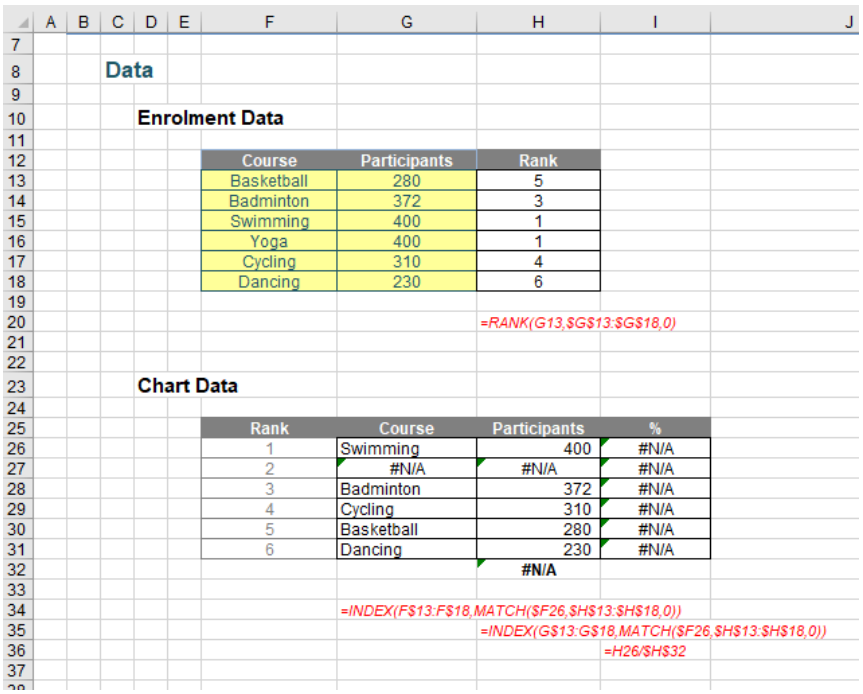
Rank	Course	Participants	%
1	Swimming	450	22.0%
2	Yoga	400	19.6%
3	Badminton	372	18.2%
4	Cycling	310	15.2%
5	Basketball	280	13.7%
6	Dancing	230	11.3%
		2,042	

=INDEX(F$13:F$18,MATCH($F26,$H$13:$H$18,0))

=INDEX(G$13:G$18,MATCH($F26,$H$13:$H$18,0))

=H26/$H$32

There is one thing that is not quite right here. Suppose the data changes and the number of participants in the Swimming course and Yoga course are equal *e.g.* they are both 400. Their ranks will be equal first and the **Chart Data** will no longer be correct.



Data

Enrolment Data

Course	Participants	Rank
Basketball	280	5
Badminton	372	3
Swimming	400	1
Yoga	400	1
Cycling	310	4
Dancing	230	6

=RANK(G13,$G$13:$G$18,0)

Chart Data

Rank	Course	Participants	%
1	Swimming	400	#N/A
2	#N/A	#N/A	#N/A
3	Badminton	372	#N/A
4	Cycling	310	#N/A
5	Basketball	280	#N/A
6	Dancing	230	#N/A
		#N/A	

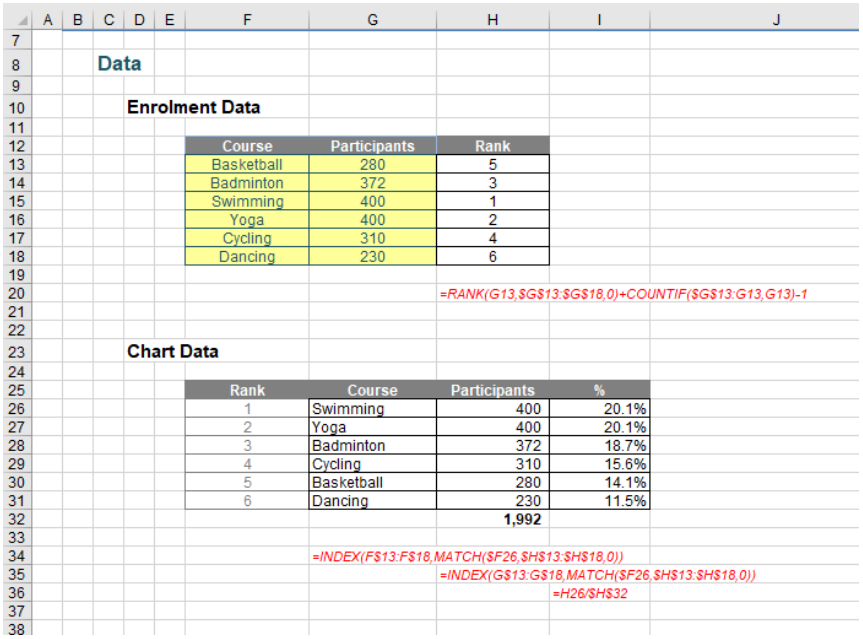
=INDEX(F$13:F$18,MATCH($F26,$H$13:$H$18,0))

=INDEX(G$13:G$18,MATCH($F26,$H$13:$H$18,0))

=H26/$H$32

Hence, we will fix the formula in cell **H13** to get the **unique rank**:

=RANK(G13,$G$13:$G$18,0)+COUNTIF($G$13:G13,G13)-1



Data

Enrolment Data

Course	Participants	Rank
Basketball	280	5
Badminton	372	3
Swimming	400	1
Yoga	400	2
Cycling	310	4
Dancing	230	6

=RANK(G13,$G$13:$G$18,0)+COUNTIF($G$13:G13,G13)-1

Chart Data

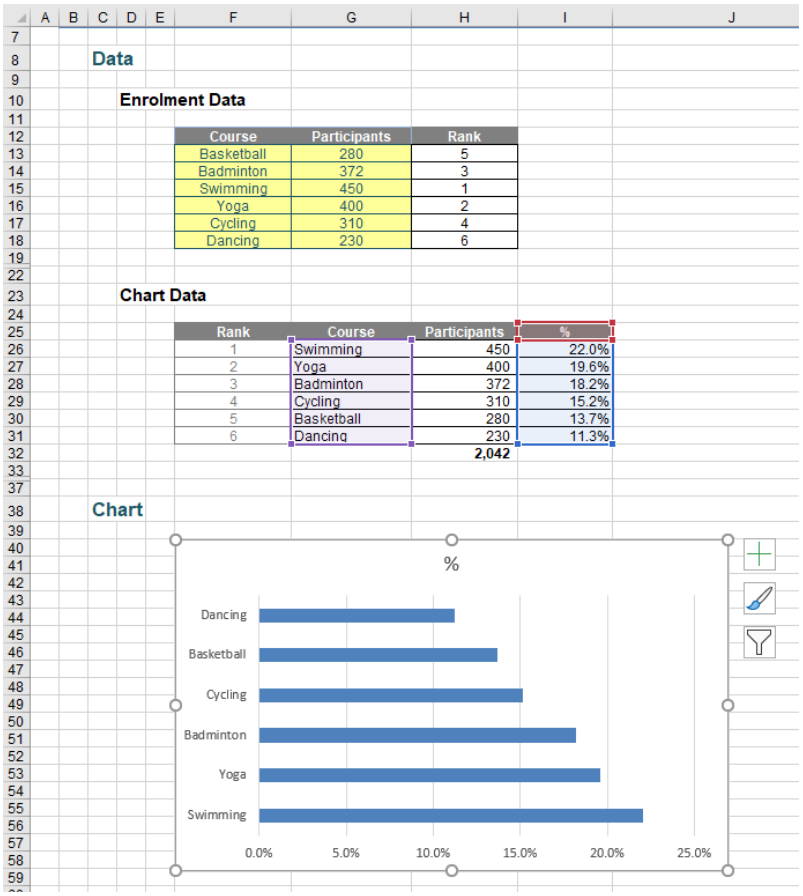
Rank	Course	Participants	%
1	Swimming	400	20.1%
2	Yoga	400	20.1%
3	Badminton	372	18.7%
4	Cycling	310	15.6%
5	Basketball	280	14.1%
6	Dancing	230	11.5%
		1,992	

=INDEX(F$13:F$18,MATCH($F26,$H$13:$H$18,0))

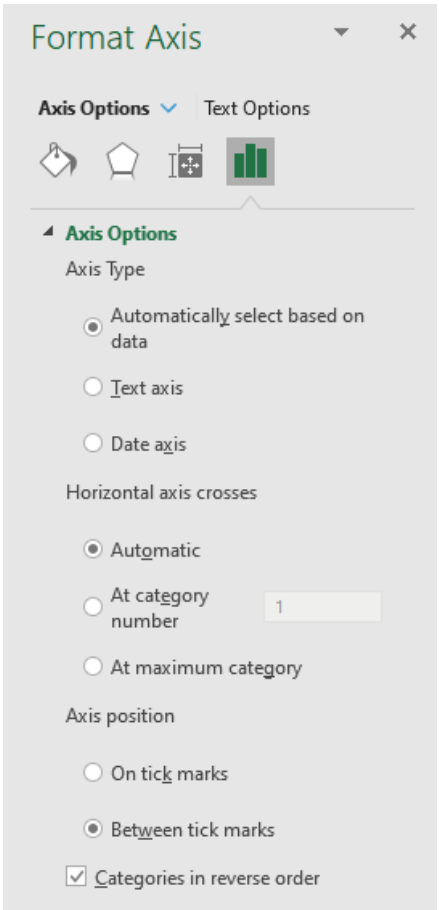
=INDEX(G$13:G$18,MATCH($F26,$H$13:$H$18,0))

=H26/$H$32

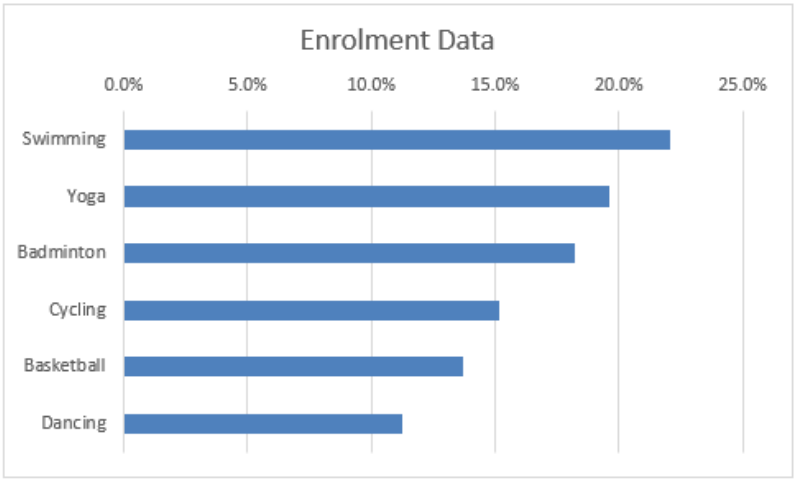
If we revert to the original data and now that we have the **Chart Data** ready, we will select the **Course** and **%** columns to create an initial Bar chart that looks like the one below.



Even though the **Chart Data** has been sorted by descending order, the Bar chart is still in ascending order. To reverse the order, right-click on the vertical axis and choose 'Format Axis'. Under the 'Format Axis' pane, check 'Categories in reverse order' option.

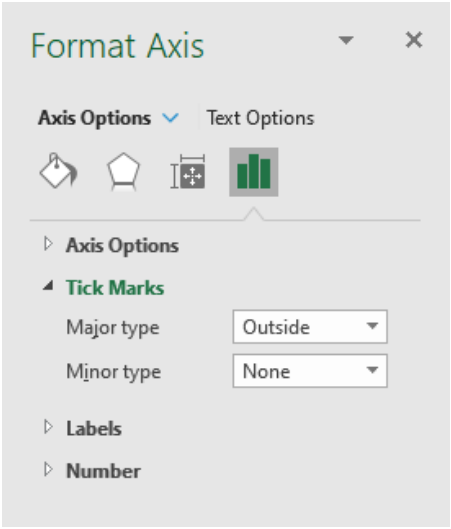


The chart is now in the correct order.



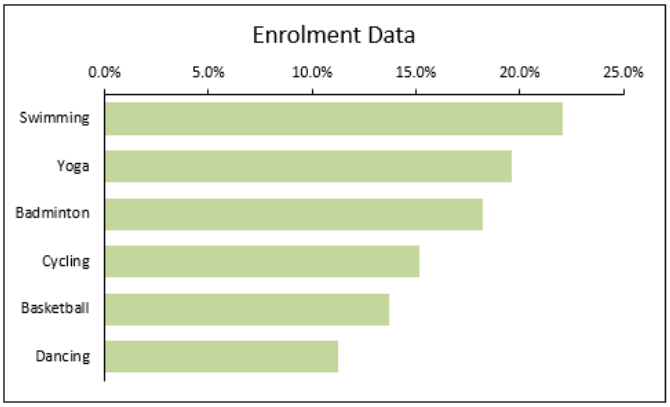
We will do some further formatting to the chart:

- remove the grid lines
- right-click on the data series and choose 'Format Data Series', and choose a colour to your liking
- under the 'Format Data Series' pane, reduce the 'Gap Width' to 50%
- right-click on the horizontal axis and choose 'Format Axis'
- under the 'Format Axis' pane, let Line be 'Solid line'
- under the 'Tick Marks' option, choose Outside for 'Major type'



- right-click on the vertical axis and choose 'Format Axis'
- under the 'Format Axis' pane, let Line be 'Solid line'
- click on the chart, and under the 'Format Chart Area' pane, let Border be 'Solid line'.

The chart should now appear like the one below:



To create a more memorable chart, we can change the chart title to tell a better story by creating a formula-driven chart title content cell, *i.e.* cell **D40** and link the chart title to this cell to create a dynamic chart title.

Data

Enrolment Data

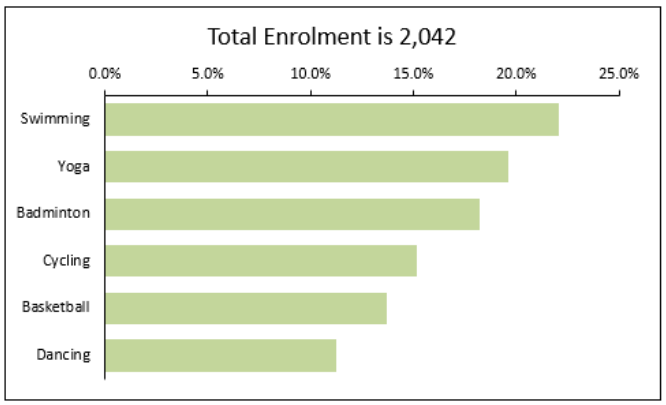
Chart Data

Rank	Course	Participants	%
1	Swimming	450	22.0%
2	Yoga	400	19.6%
3	Badminton	372	18.2%
4	Cycling	310	15.2%
5	Basketball	280	13.7%
6	Dancing	230	11.3%
		2,042	

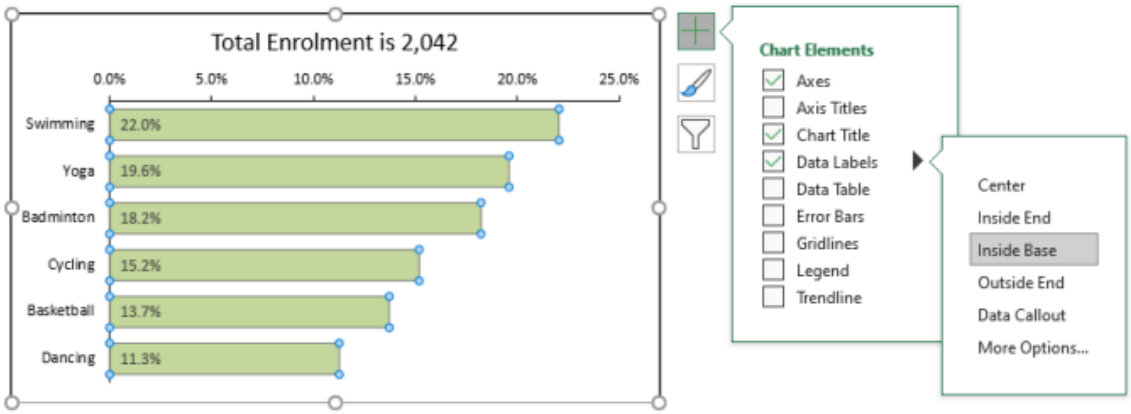
Chart

Total Enrolment is 2,042

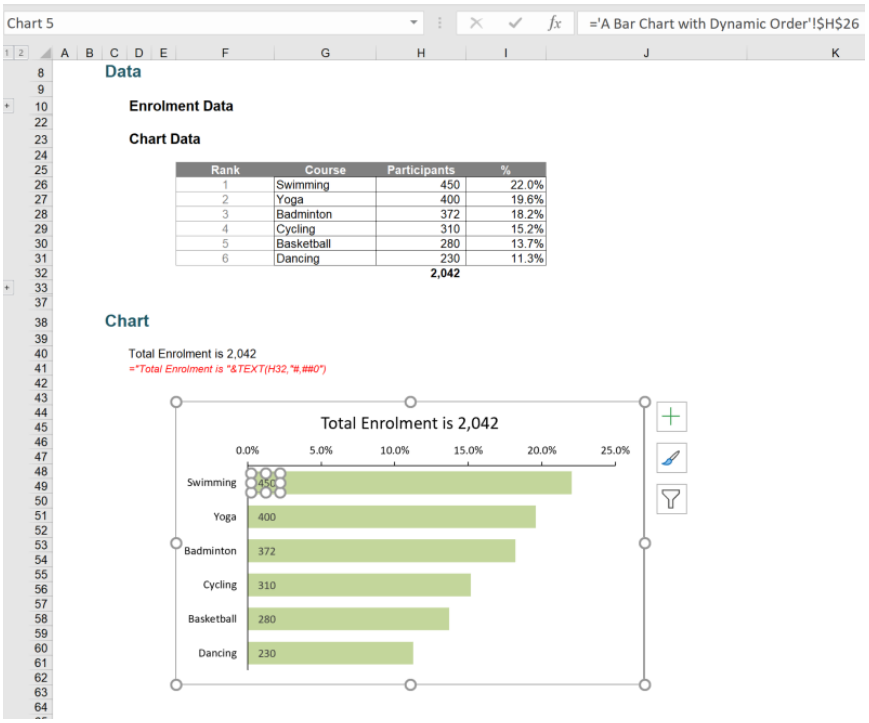
="Total Enrolment is "&TEXT(H32,"#,##0")



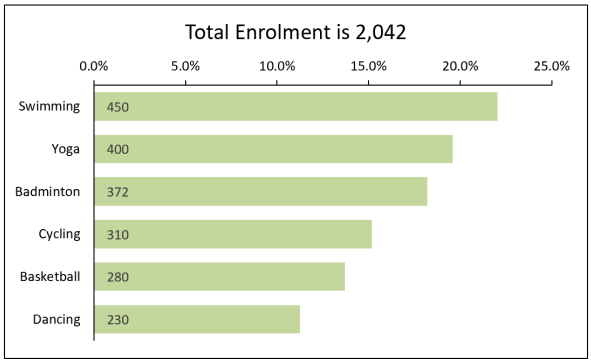
We can also add data labels to the chart:



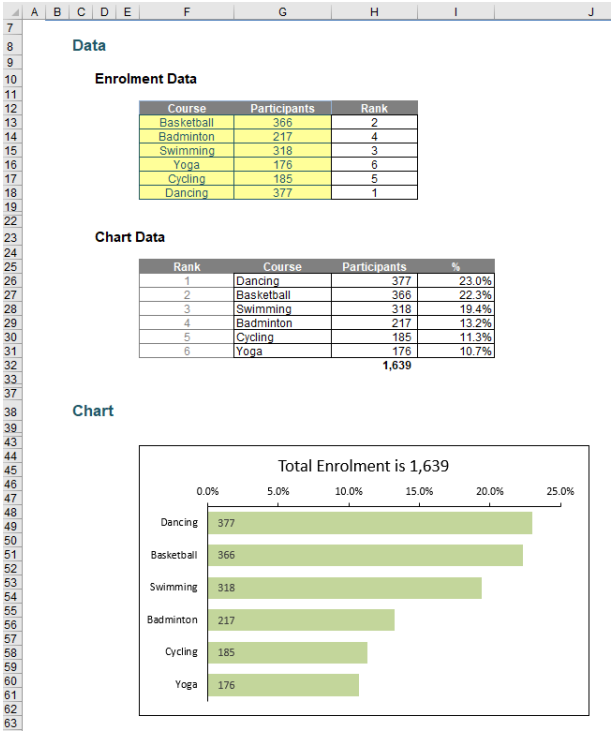
Instead of using data labels as a percentage (while the chart is already based on the proportion of course participants), we can point the data label to the number of participant column, *i.e.* cells **H26:H32**.



The chart should now present like the one below:



If we randomly change the enrolment data, all of the chart elements will be updated.



More next month.

Power Pivot Principles

*We continue our series on the Excel COM add-in, Power Pivot. This month, we look at the **REPLACE** function.*

The **REPLACE** function is used in creating custom columns in the 'Power Pivot for Excel' window, to create new groupings or categories for datasets. The **REPLACE** function replaces part of a text string in each row in a column, based upon the number of characters specified by the user, with a different text string.

The **REPLACE** function uses the following syntax to operate:

REPLACE(old_text, start_num, num_characters, new_text)

- the **old_text** parameter is the string of text that contains the characters that we want to replace, this can also be a reference to a column that contains text
- the **start_num** parameter is the position of the first character in the **old_text** that we wish to replace with **new_text**
- the **num_chars** parameter is the number of characters that we want to replace
- the **new_text** parameter is the text that is going to replace the **old_text**.

Let's move on to an example. Here, we are going to use the following extremely complicated dataset:

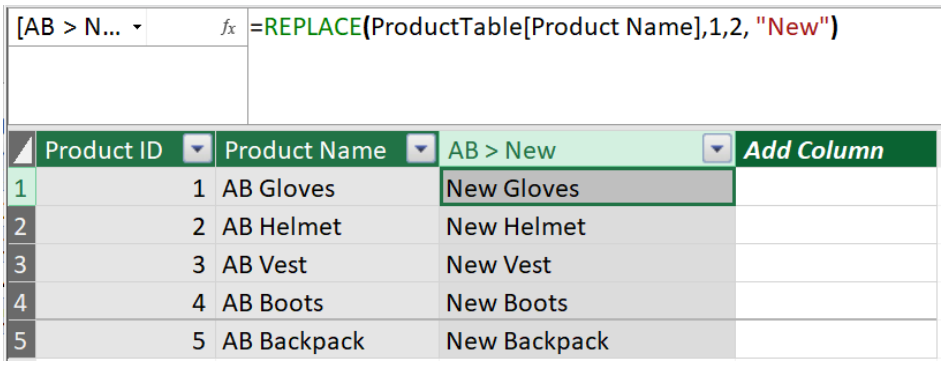
Product ID	Product Name
1	AB Gloves
2	AB Helmet
3	AB Vest
4	AB Boots
5	AB Backpack

Picture that an intern has accidentally typed in "AB" in front of all the product names, instead of "New". Sure, we can fix this in Excel, but let's do it in **DAX**. We can use the **REPLACE** function in **DAX** to create a calculated column that will correct this step.

After adding the data table to our data set in Power Pivot, we can create a calculated column with the following **DAX** function:

=REPLACE(ProductTable[Product Name],1,2, "New")

As a side note, we have entered '1' as the **start_num** as "AB" is in the first character position in the "AB Gloves" text string. If we wanted to replace "Gloves" we would specify '4' as the **start_num**: this is because we count each letter and blank in the text string as a character position.



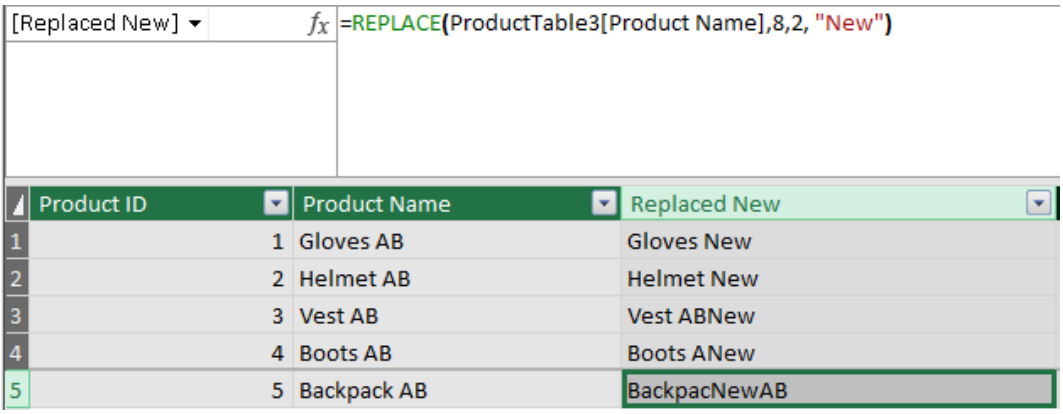
Problem solved! The **REPLACE** function works well with replacing text strings "AB" that all begin in the same spot in the text field. What if the AB were at the end of the text string?

Product ID	Product Name
1	Gloves AB
2	Helmet AB
3	Vest AB
4	Boots AB
5	Backpack AB

We can specify the **start_num** as '8' since "AB" begins on the eighth character position in "Gloves AB".

=REPLACE(ProductTable3[Product Name],8,2, "New")

However, this does not work very well as AB begins in different positions in each row, *viz.*



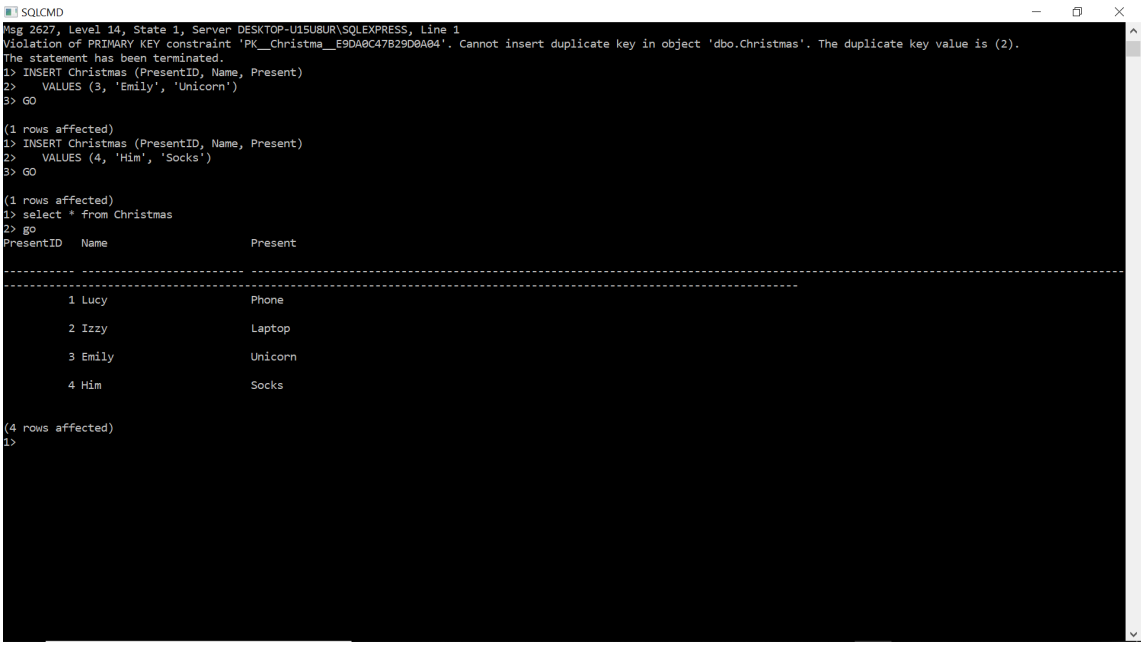
	Product ID	Product Name	Replaced New
1	1	Gloves AB	Gloves New
2	2	Helmet AB	Helmet New
3	3	Vest AB	Vest ABNew
4	4	Boots AB	Boots ANew
5	5	Backpack AB	BackpacNewAB

Maybe we need a different function – but more on that next month…

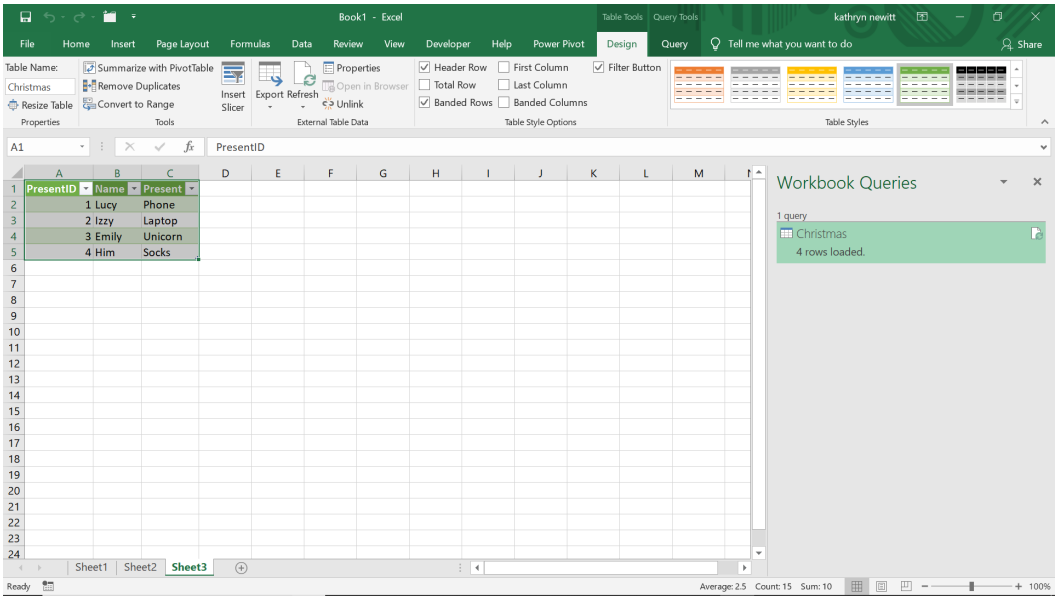
Power Query Pointers

Each month we'll reproduce one of our articles on Power Query (Excel 2010 and 2013) / Get & Transform (Office 365, Excel 2016 and 2019) from www.sumproduct.com/blog. If you wish to read more in the meantime, simply check out our Blog section each Wednesday. This month, we consider updating data in a SQL database.

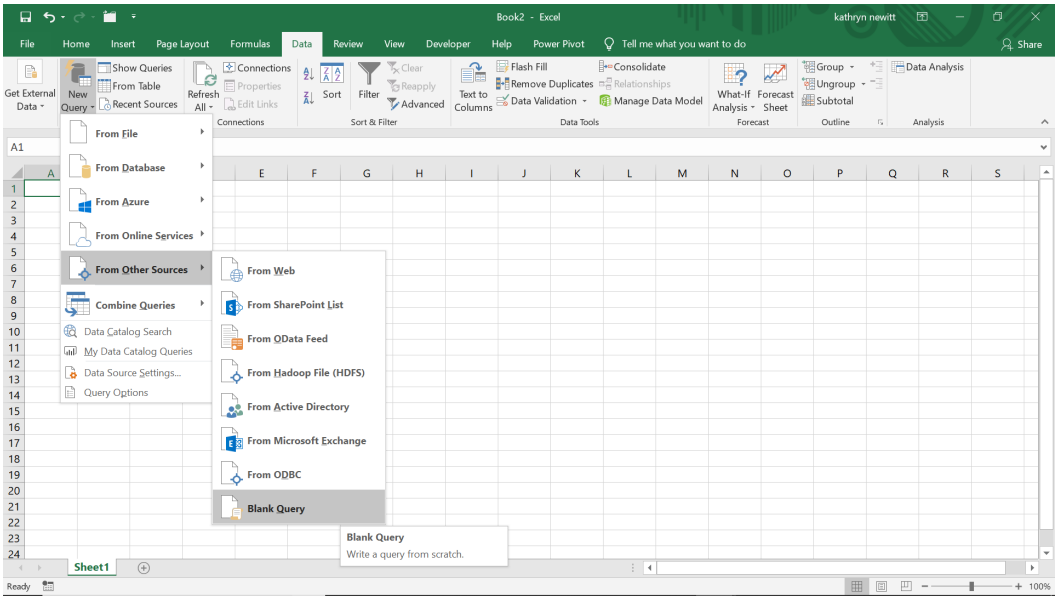
This month, we'll start with a SQL database where we may create a new simple table:



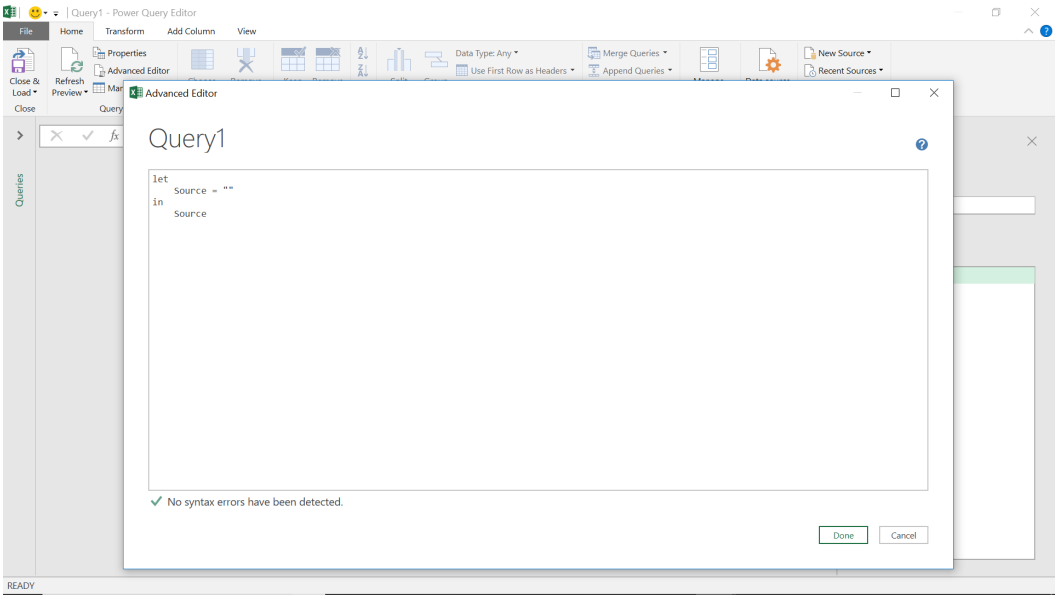
We will create a new Excel workbook, and load the data from this SQL database to test the connections.



We may create a blank query, where we can enter the **M** code to update our database.



Once in our new Blank Query, we can call the Advanced Editor.



Let's enter **M** code that will update a row in our database.

In our very unseasonal example, we intend to change the present for Lucy…

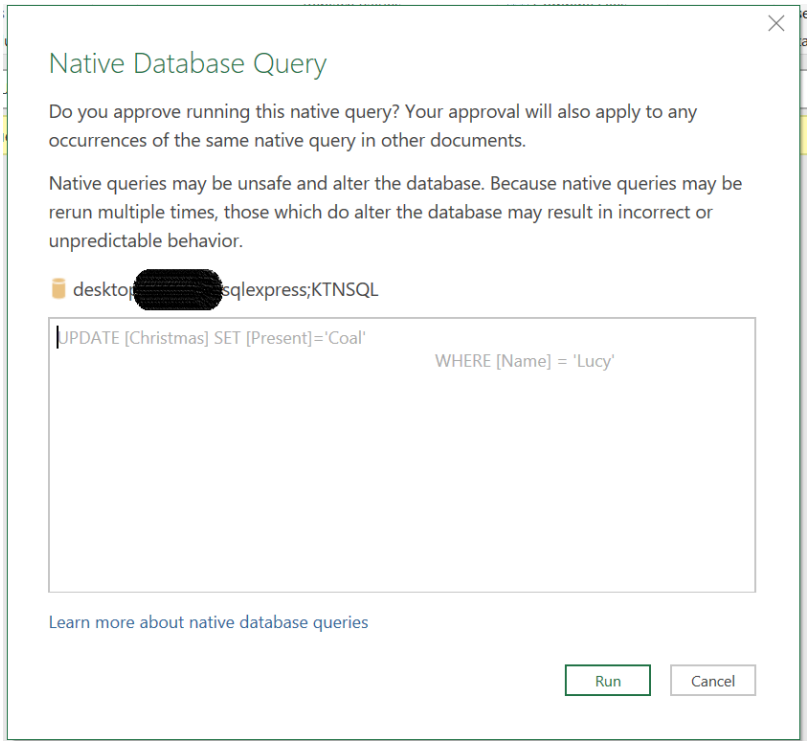
```
let
  fix_christmas = Sql.Database("DESKTOP-nnnnn\SQLEXPRESS","KTNSQL",[Query="UPDATE [Christmas] SET [Present]='Coal'
                                WHERE [Name] = 'Lucy'"])
in
  fix_christmas
```

This uses the **M** function **Sql.Database()**:

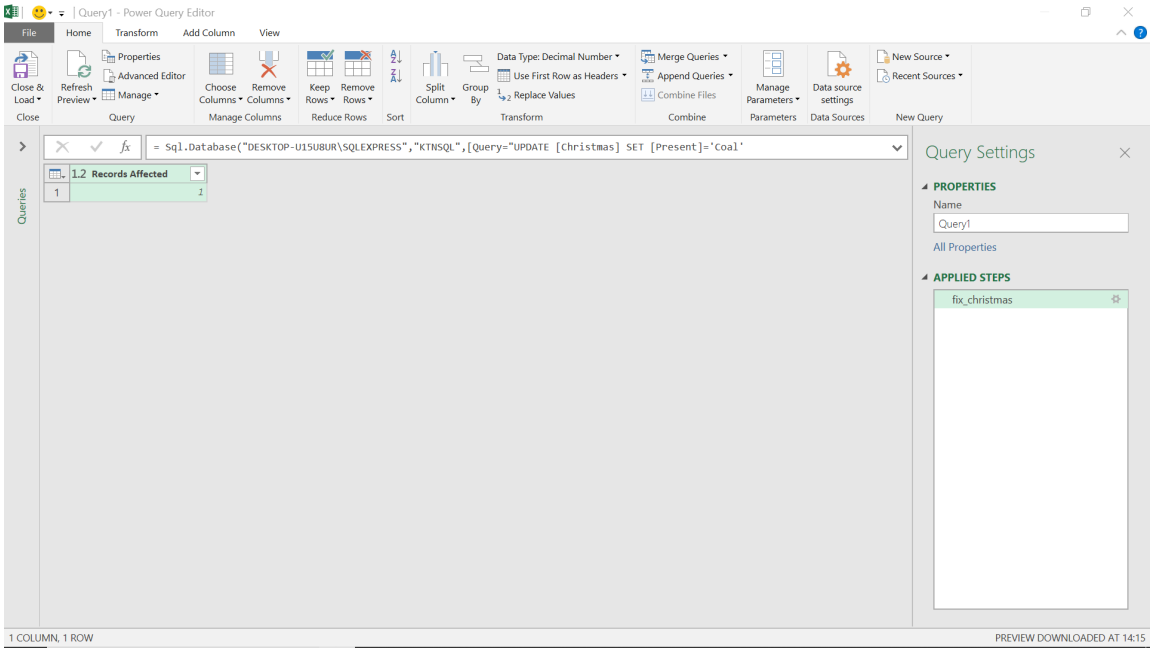
Sql.Database("server" as text, **"database"** as text, **optional options** as nullable record**)** as table

This function allows manipulation of an SQL database. In this case, we have provided a server name (which we have amended here for privacy / security purposes!), a database name 'KTNSQL' and an SQL query to update a row on my database.

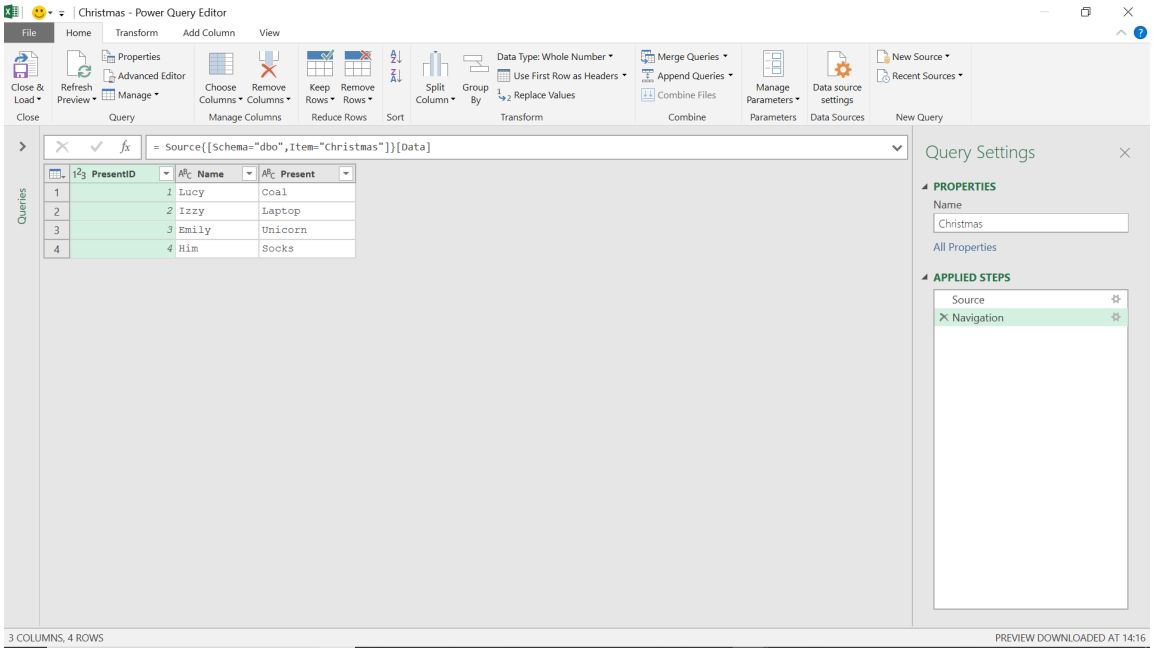
Since we're updating the database, Power Query needs permission:



Let's allow the query to run.



We can check the data by refreshing the 'Christmas' query to extract data from the table:



We have successfully updated a row on my database.

This is what programmers refer to as 'an undocumented feature'! This means that it can only be used with caution. The help page on **Sql.Database()** does not mention updating the database, and it is entirely possible that this feature could be removed in the future. On the other hand, it could be developed into a supported function and add another string to the bow that is Power Query.

Currently, it is not the most efficient way to update a database, but it's interesting to know what's possible!

Until next month.

Power BI Updates

We've waited and waited and waited this month, but no news is forthcoming from Redmond Towers as at our printing deadline. Perhaps they are planning to cancel Power BI..?

Nah, I don't think so. Enjoy the break and we will assume they will advise of updates (maybe a double load?) next month.

New Features for Excel

Copilot in Excel with Python is now available on Windows platforms for all enterprise and consumer users. Also starting now, Excel users on Windows and web can get to Copilot faster from anywhere on the grid (aren't there enough infernal prompts already!?). Furthermore, Copilot can now bring web and organisational data into your spreadsheets—available for Windows and Mac Insider users.

For completists, here is that summarised list:

Excel for Windows

- Copilot in Excel with Python

Excel for Windows and web

- Get to Copilot in Excel faster

Excel for Windows and Mac

- Search and import data with Copilot in Excel (Insiders).

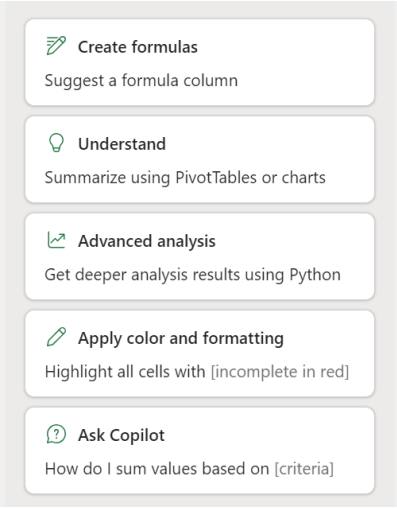
Let's get started.

Copilot in Excel with Python

Copilot in Excel with Python is a tool that leverages the capabilities of artificial intelligence (AI) to get deeper insights from your data and take advantage of Python in Excel without being a Python or Excel expert.

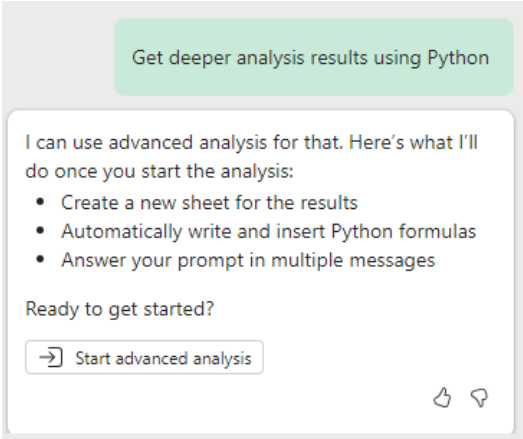
To get started with Copilot in Excel with Python on Windows platforms, follow these steps:

1. Open Excel on Windows and load your data
2. Navigate to the Home tab, then select Copilot
3. Select the 'Advanced analysis' card in the task pane that opens



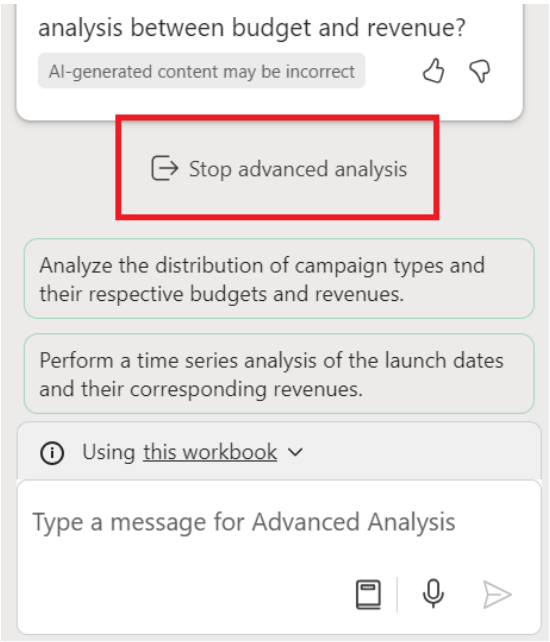
4. Alternatively, in the task pane you can use prompts like:
    - "Analyse this data using Python"
    - "Run advanced analysis using Python"

5. Select 'Start advanced analysis' to enter the advanced analysis mode. Once in this mode, a new sheet is created for your analysis and Copilot inserts any Python code needed for your analysis. You can ask follow-up questions, request explanations or use the suggested prompts that appear above the chat input



Note: Copilot in Excel is compliant with data protection standards

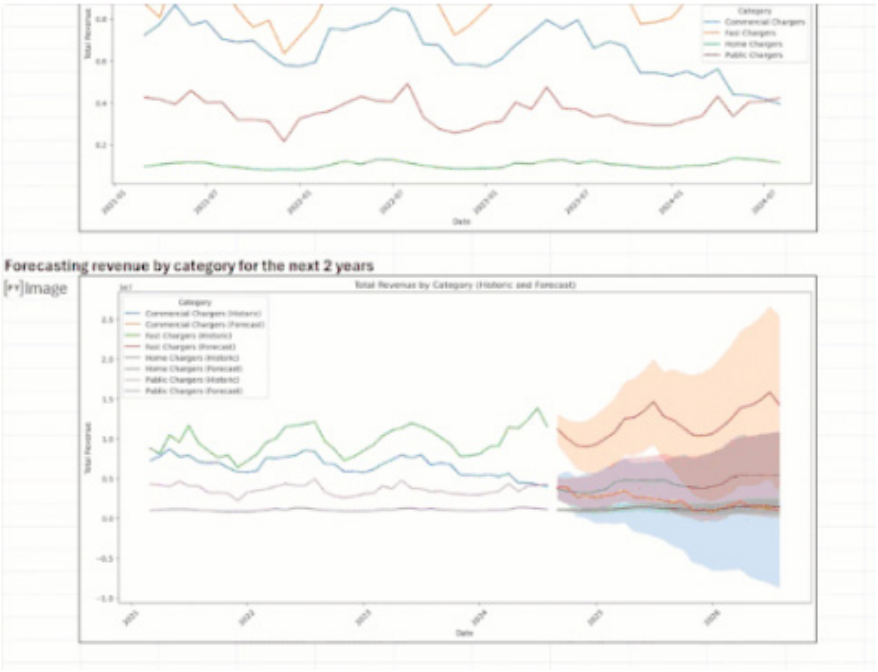
6. To exit advanced analysis mode, select 'Stop advanced analysis'. This takes Copilot out of the advanced analysis mode and back to its regular mode for other Copilot skills.



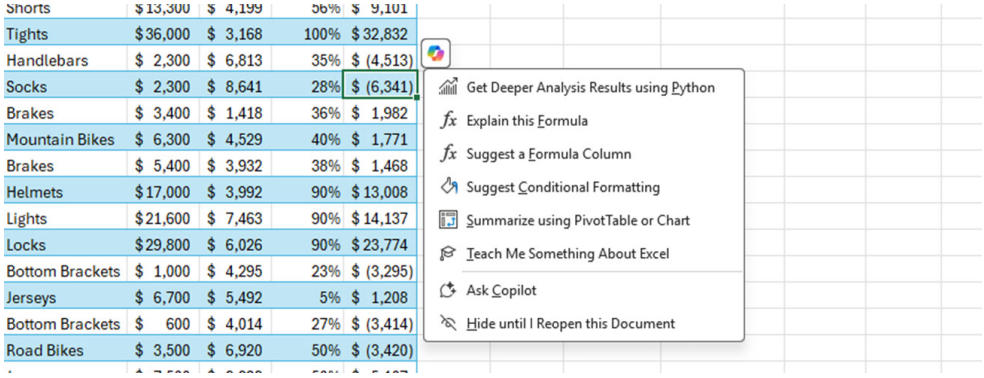
Copilot in Excel with Python is available worldwide on Windows platform for all enterprise and consumer users in English, Simplified Chinese, French, German, Italian, Japanese, Brazilian Portuguese and Spanish languages with the following version requirements:

- Current Channel starting with Version 2409 (Build 16.0.18025.00000)
- Monthly Enterprise Channel starting with Version 2410 (Build 16.0.18227.00000).

It should be noted that you must have Python in Excel available to access this feature. Support for other platforms, and additional languages will be included in future iterations.

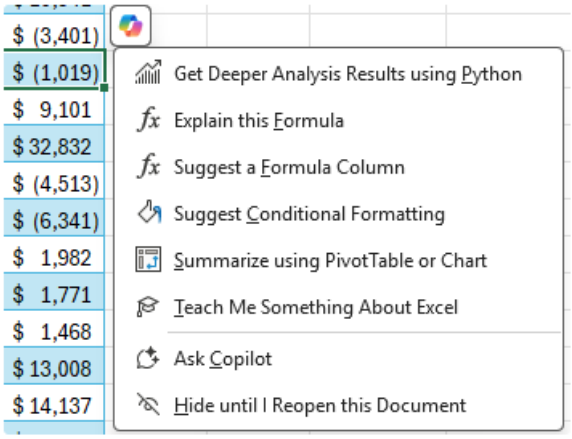


Get to Copilot in Excel faster

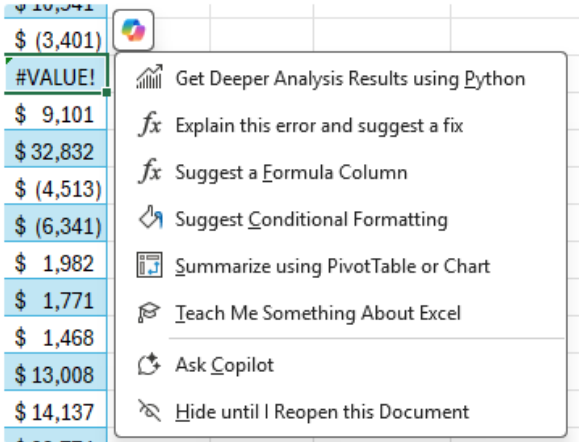


To help you use Copilot more easily in Excel for Windows and web, Microsoft has added a new Copilot button that lets you access Copilot from anywhere on the grid, so that you can stay in your flow. This button will appear next to your cell selection and can be activated at any time to see a list of suggestions.

Selecting any of these options will open the Copilot side pane and automatically run that prompt as if you had typed it manually. If you don't need the Copilot icon, you can hide it until the file is reopened:



It should be noted that depending on whether the selected cell has a valid formula or a formulaic error, you will see the corresponding option appear in the menu.



Very shortly, Microsoft will be adding a setting that will give you more control over when the Copilot icon is visible, and this should be available in the coming weeks. Here's a sneak peek at how it will look (final design might change slightly):

Access to Copilot on the grid is gradually rolling out to consumer and business users, both in Excel for Windows (Build 16.0.18324.20000 or later) and Excel for the web.

Search and import data with Copilot in Excel (Insiders)

In Excel for Windows and Mac, Copilot in Excel can now reference Word, Excel, PowerPoint and PDF files, making the entire data collection and analysis process faster, simpler and more intuitive. This new capability works for gathering information from sources such as web content, internal content, organisation(al) information or data from another Excel file.

Data analysis is a critical yet challenging task for many of us, as creating and preparing data, then gaining insights from it and conducting advanced analyses, can be time-consuming and complex.

This is where Copilot in Excel can help. This new capability works for gathering information such as:

- **web content:** you can now search the web directly within Copilot in Excel to find public information like dates, statistics and more, without disrupting your workflow. For instance, you can look up a list of countries and their exchange rates and easily paste these results into a table

- **internal content:** for example, you can ask Copilot in Excel to list the announcements from a newsletter drafted in Word and Copilot will respond with a list that you can insert into a new spreadsheet

- **organisational information:** you can ask Copilot for all the employees who report to a specific manager and insert this list into a spreadsheet

- **data from another Excel file:** one of Microsoft's top user requests is to import data from another Excel file. This is possible with Power Query, but doing so takes time and knowledge – not to mention, searching for the right file can be frustrating. Now you can ask Copilot for help finding and importing a table from an Excel file using everyday language.

  For example, say you want to add budget details from another Excel spreadsheet to your sales team table: you can describe the data that you need and Copilot will help you import it. Since this import is powered by Power Query, it brings precise data directly from your organisation's data sources as a refreshable connection. As data is updated in the budget file, it also updates in your sales team table.

To operate this feature:

1. On the Home tab in Excel, select Copilot
2. Ask Copilot a question about the data you are looking for
3. Ask follow-up questions to Copilot's responses until you're happy with the results.

You should note that for results that are from external data, make sure you see reference(s) at the end of the response.

There are some known limitations presently:

- importing refreshable data is only supported for Excel files with Tables, stored on your SharePoint or OneDrive
- addressing workbook data and asking for external data simultaneously is not fully supported yet.

To use this feature, you must have:

- a Copilot licence
- a stable internet connection
- web search enabled (for web searching).

If Copilot responds that it can't access your organisational data, verify if 'Restricted SharePoint Search' is turned on (for Enterprise search).

This feature is available to Beta Channel users who meet the requirements outlined above and are running:

- Windows: Version 2503 (Build 17729.20000) or later
- Mac: Version 16.87 (Build 24053110) or later.

This update to Copilot in Excel will become available to Excel for the web users in the near future.

Until next month.

The A to Z of Excel Functions: PERMUT

This function returns the number of permutations for a given number of objects that can be selected from number objects. A permutation is any set or subset of objects or events where internal order is significant (*i.e.* order is important). Permutations are different from combinations, for which the internal order is not significant (*i.e.* order is unimportant).

It is defined as follows:

$$ {}_nP_r = \frac{n!}{(n-r)!} $$

${}_nP_r$ = permutation

n = total number of objects

r = number of objects selected

For example, selecting two numbers out of the set {1, 2, 3, 4} where order is important gives you the following pairs: {1, 2}, {1, 3}, {1, 4}, {2, 1}, {2, 3}, {2, 4}, {3, 1}, {3, 2}, {3, 4}, {4, 1}, {4, 2} and {4, 3}, which is 12 ordered pairs.

You can clearly see you could use this function for bingo-style probability calculations and realise why you will never be rich. (Do note this example: bingo is appropriate here since it considers when you get the winning final number; for a lottery, you would not care which order the numbers were selected as long as they are picked.)

The syntax of the **PERMUT** function is as follows:

=PERMUT(number, number_chosen)

It contains two [2] arguments:

- **number:** this is required and represents an integer that describes the number of objects
- **number_chosen:** this is also required and denotes an integer that describes the number of objects in each permutation.

It should be noted that:

- both arguments are truncated to integers
- if **number** or **number_chosen** is nonnumeric, **PERMUT** returns the *#VALUE!* error value
- if **number** ≤ 0 or if **number_chosen** < 0, **PERMUT** returns the *#NUM!* error value
- if **number** < **number_chosen**, **PERMUT** returns the *#NUM!* error value.

Please see our examples below:

	A	B	C
1	Independent values	Dependent values	
2	4	Number of objects	
3	2	Number of objects in each permutation	
4			
5			
6	Formula	Description	Result
7	=PERMUT(A2,A3)	Permutations possible for the arguments specified in cells A2:A3	12
8	=PERMUT(100, 3)	Permutations possible for a group of 100 objects where three [3] are chosen	970,200

The A to Z of Excel Functions: PERMUTATIONA

This function returns the number of permutations for a given number of objects that can be selected from number objects allowing for repetitions (*i.e.* items selected are then replaced). A permutation is any set or subset of objects or events where internal order is significant (*i.e.* order is important). Permutations are different from combinations, for which the internal order is not significant (*i.e.* order is unimportant).

It is defined as follows:

$$\textbf{PERMUTATIONA} = \text{Total}^{\text{Chosen}}$$

For example, selecting two numbers with replacement out of the set {1, 2, 3, 4} where order is important gives you the following pairs: {1, 1}, {1, 2}, {1, 3}, {1, 4}, {2, 1}, {2, 2}, {2, 3}, {2, 4}, {3, 1}, {3, 2}, {3, 3}, {3, 4}, {4, 1}, {4, 2}, {4, 3} and {4, 4}, which is 16 ordered pairs.

The syntax of the **PERMUTATIONA** function is as follows:

=PERMUTATIONA(number, number_chosen)

It contains two arguments:

- **number:** this is required and represents an integer that describes the number of objects
- **number_chosen:** this is also required and denotes an integer that describes the number of objects in each permutation.

It should be noted that:

- both arguments are truncated to integers
- if numeric arguments are values that are not valid, *e.g.* when the total number is zero [0] and the chosen number is larger than zero [0], **PERMUTATIONA** returns the *#NUM*! error value
- if numeric arguments use data types that are nonnumeric, **PERMUTATIONA** returns the *#VALUE!* error value.

Please see our final examples for this month below:

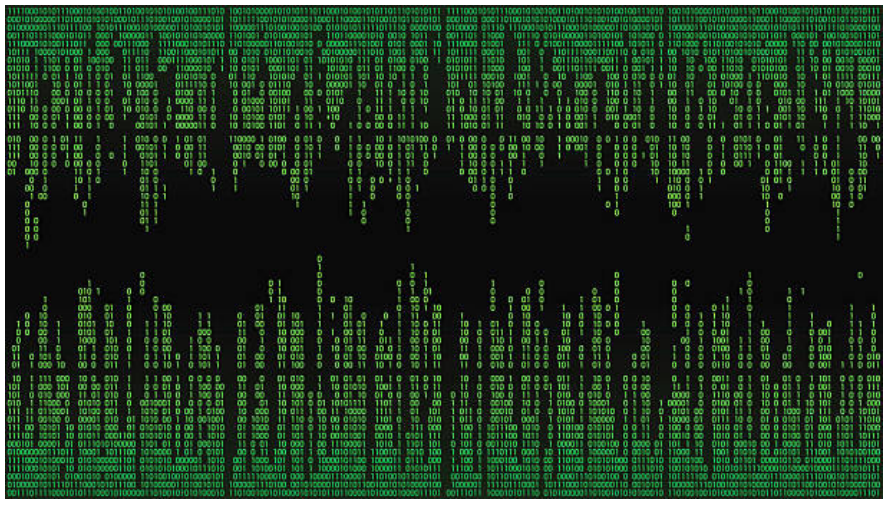
	A	B	C
1	Independent values	Dependent values	
2	4	Number of objects	
3	2	Number of objects in each permutation	
4			
5			
6	Formula	Description	Result
7	*=PERMUTATIONA(A2,A3)*	Permutations with replacement possible for the arguments specified in cells **A2:A3**	16
8	*=PERMUTATIONA(100, 3)*	Permutations with replacement possible for a group of 100 objects where three [3] are chosen	1,000,000

More Excel Functions next month.

Beat the Boredom Suggested Solution

Earlier in this newsletter, we set a fairly straightforward challenge. The challenge this month was to create a line in a Matrix visualisation in Power BI. This should be performed only using the tools within Power BI, not by drawing a Shape in the report.

The Challenge



In Power BI there is a visualisation called the Matrix visualisation. We can use it to display numerical values over several time periods:

Year	2021									
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct
Total Sales	1,000.00	1,100.00	1,150.00	950.00	900.00	800.00	980.00	600.00	1,050.00	1,200.00
Total COGS	450.00	525.00	537.50	487.50	475.00	450.00	370.00	275.00	387.50	550.00
Gross Profit	550.00	575.00	612.50	462.50	425.00	350.00	610.00	325.00	662.50	650.00

Each line item is a measure. The challenge here was to make the **Gross Profit** measure stand out more by inserting lines into the Matrix visualisation like so:

Year	2021									
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct
Total Sales	1,000.00	1,100.00	1,150.00	950.00	900.00	800.00	980.00	600.00	1,050.00	1,200.00
Total COGS	450.00	525.00	537.50	487.50	475.00	450.00	370.00	275.00	387.50	550.00
Gross Profit	550.00	575.00	612.50	462.50	425.00	350.00	610.00	325.00	662.50	650.00

We did not simply draw a line on top of the visualisation. We can expand the visualisation by breaking the **Total COGS** measure down to **Direct Labour** and **Direct Materials**:

Year	2021									
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct
Total Sales	1,000.00	1,100.00	1,150.00	950.00	900.00	800.00	980.00	600.00	1,050.00	1,200.00
COGS										
Direct Labour	200.00	250.00	250.00	250.00	250.00	250.00	125.00	125.00	125.00	250.00
Direct Materials	250.00	275.00	287.50	237.50	225.00	200.00	245.00	150.00	262.50	300.00
Total COGS	450.00	525.00	537.50	487.50	475.00	450.00	370.00	275.00	387.50	550.00
Gross Profit	550.00	575.00	612.50	462.50	425.00	350.00	610.00	325.00	662.50	650.00

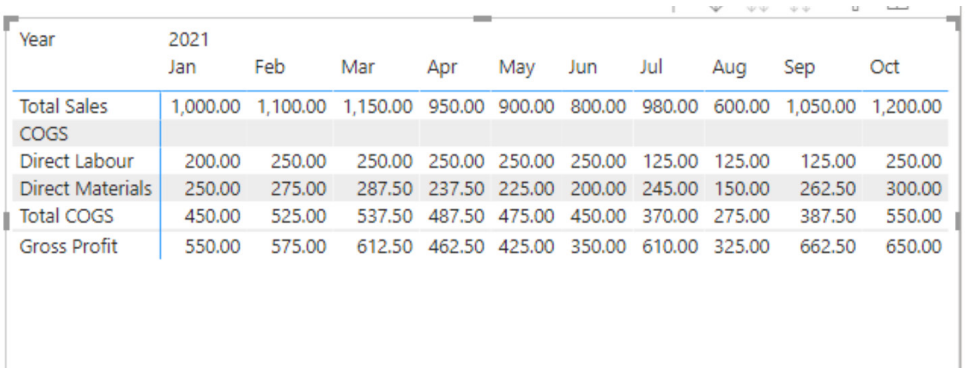
The lines move *automatically.*

Suggested Solution

The first step here is to create a new measure, in this case we are going to enter the following DAX code into the measure formula bar:

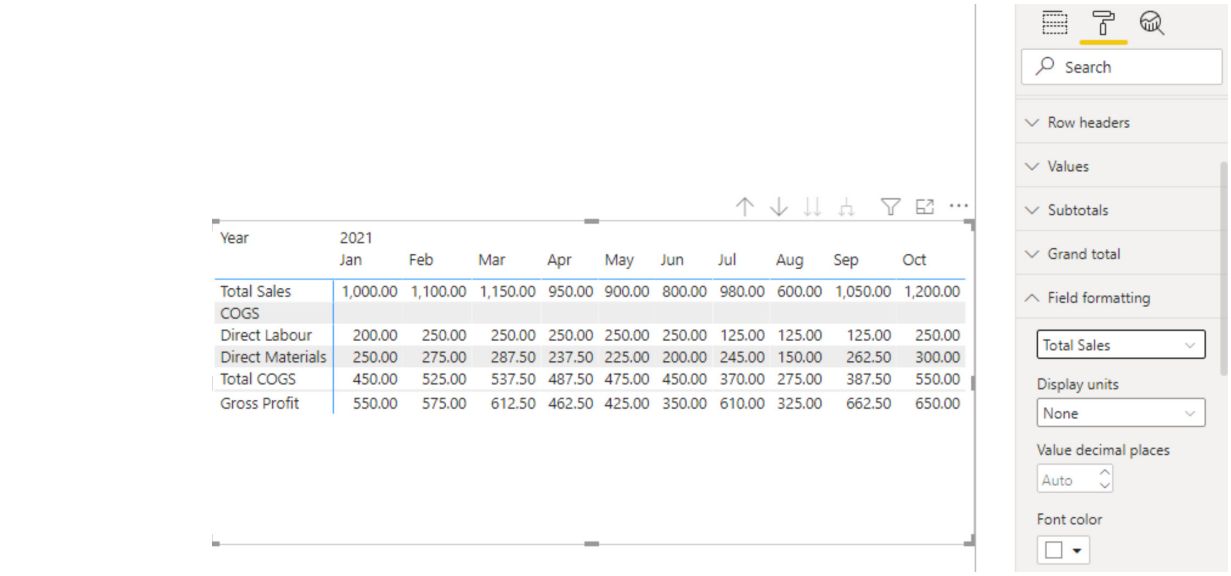
*** = " "**

We use the asterisk in this example, because when shown on a visualisation, the Asterisk defaults to a blank space. That's a nice trick to know. For example, if we place the newly created Asterisk measure in between the **Total COGS** and the **Gross Profit** measure we get the following result:

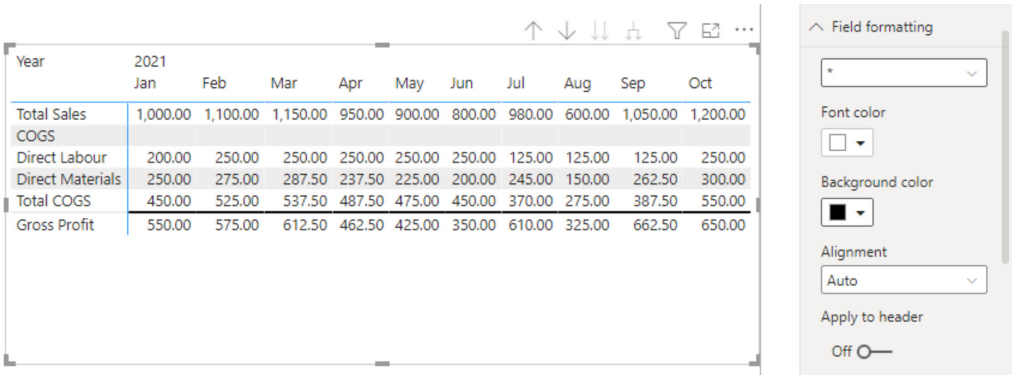


Year	2021									
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct
Total Sales	1,000.00	1,100.00	1,150.00	950.00	900.00	800.00	980.00	600.00	1,050.00	1,200.00
COGS										
Direct Labour	200.00	250.00	250.00	250.00	250.00	250.00	125.00	125.00	125.00	250.00
Direct Materials	250.00	275.00	287.50	237.50	225.00	200.00	245.00	150.00	262.50	300.00
Total COGS	450.00	525.00	537.50	487.50	475.00	450.00	370.00	275.00	387.50	550.00
Gross Profit	550.00	575.00	612.50	462.50	425.00	350.00	610.00	325.00	662.50	650.00

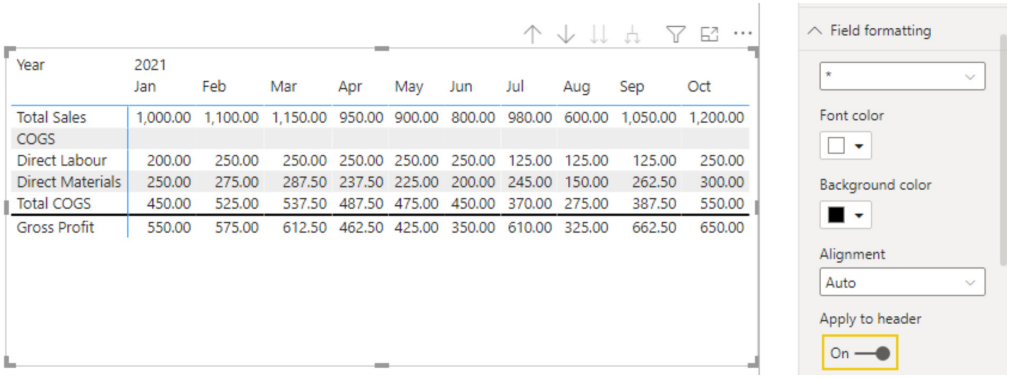
Looking at the visualisation it is currently a grey line. We can change that by navigating to the Format tab, and expanding the 'Field formatting' section:



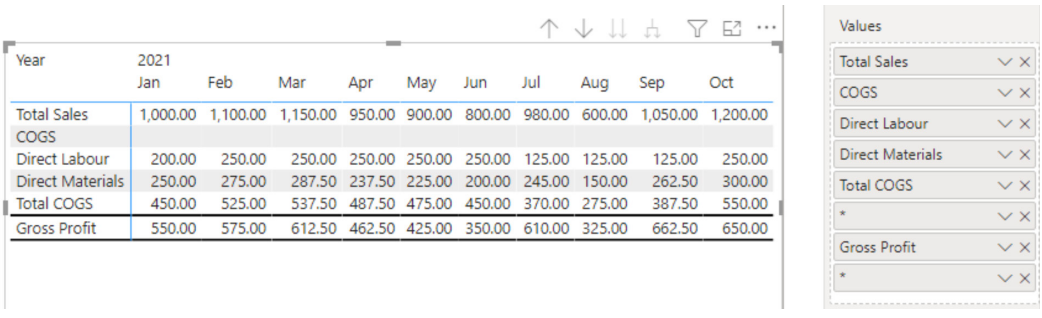
From here, we change the formatting of the asterisk (*) measure, with the trick being to change the 'Background color' to black:



The next step is to toggle the 'Apply to header' option to On.



To add the line below the **Gross Profit** measure, we simply add another Asterisk measure below the **Gross Profit** measure in the Values area:



Did you have a better solution? Let us know, we'd be glad to hear if there are better ways to work around this problem!

Upcoming SumProduct Training Courses

Location	Course	Course Date	Local TIme	UTC	Duration
Virtual (Australia)	ChatGPT Part 2	7 April 2025	09:00 - 12:30 AEDT	6 Apr 2025 23:00 UTC - 7 Apr 2025 02:30 UTC	1 Day
Sydney Australia	Power Pivot, Power Query and Power BI	10 April 2025 - 11 April 2025	09:00 - 17:00 AEDT	9 April 2025 23:00 UTC - 11 April 2025 07:00 UTC	2 Days

Key Strokes

Each newsletter, we'd like to introduce you to useful keystrokes you may or may not be aware of. This time, we thought we would regain **Alt**er the alphabet, but be warned – it's a bit of a blast from the past:

Keystroke	What it does
CTRL + ALT + L	Reapply Sort / Filter
CTRL + ALT + V	Paste Special dialog
CTRL + ALT + SHIFT + A	Adds French accent aigu
CTRL + ALT + SHIFT + E	Adds French accent aigu
CTRL + ALT + SHIFT + I	Adds French accent aigu
CTRL + ALT + SHIFT + O	Adds French accent aigu
CTRL + ALT + SHIFT + U	Adds French accent aigu

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.

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
www.sumproduct.com
+61 3 9020 2071