

Sum Product

NEWSLETTER #90 - May 2020

www.sumproduct.com | www.sumproduct.com/thought



We hope you are all safe and well. The COVID-19 pandemic may be forcing us all into social distancing and putting up some barriers, but there's no way SumProduct is going into lockdown!

Our latest bumper edition of the newsletter has all the usual stuff: Visual Basics, Power Pivot Principles, Power Query Pointers, Power BI (Service and Mobile) updates, Keyboard Shortcuts *et al* – but that's not all!

We share some tips on working from home, and swamp you with news on the Kindle version of our book, the Microsoft Visio Data Visualizer (*sic*) add-in, Office Scripts becoming more accessible, how to get Money in Excel, new smart templates and Data Types – and even teasing news on this year's Excel Summit. There may be no physical initiative, but we're going Virtually Global!

And for the time being, we're adding a new feature: Beat the Boredom, where you can test your Excel skills all in the name of a bit of fun! But don't stop there if you want to keep learning – all our training courses are to become more accessible online too. Details inside.

Stay safe, stay happy, stay healthy.

Until next month.

Liam Bastick, Managing Director, SumProduct



Coronavirus: How to Work from Home

The world is going through a bizarre time, much of the globe is under lockdown. In the midst of the pandemic, except for essential services, many companies are practicing voluntary or mandatory working-from-home. That means, to a lot of us, an unusual challenge. SumProduct has been a very agile firm where we have been working remotely from all sorts

of different locations for quite some time, be it at airports, hotel rooms, airplanes and home. We thought that it would be useful to share some habits and tips that we have adopted to ensure productivity and, in the meantime, personal well-being.

1: Location, location, location!

The surroundings can shape the way we think and feel, hence, setting up a proper workstation at home should be at first priority. You should have a designated place with a decent desk and chair, all machines connected, well-lit, and maybe, a touch of green, so that you can feel focused and comfortable at the same time (so psychologists tell me). Also, you need to make sure this workstation is the space just for yourself, not an area to share with anyone you are sharing the house

with, and most important, not to get any distractions from your family when you are working.

Working from home does not mean staying at home and working, lying on the bed with the computer on your lap. There may be no one watching you, but at the end of the day, your work output and the fulfillment feeling is what makes it all meaningful.

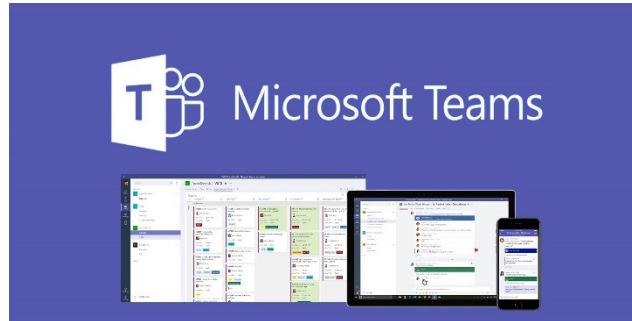


Workstation at home

2: Don't lose touch

Let's make self-isolation physical only. More and more companies are moving online, hosting virtual conferences. Our company SumProduct has offices in Sydney, Melbourne, London and New York, and we work with clients around the globe. Hence, smooth and effective communication is one thing we must always make sure of. As we move to work remotely, everyone in the team needs to stay connected. It is important to keep the team informed about the working progress and potential issues in relation to daily tasks.

Microsoft Teams is the primary tool that we use. There are others (e.g. Zoom). Teams has conference and chat tools, and is connected to our emails and calendar. The Teams channel makes it easier to work together. For example, our team is working together on a project, so we create a channel to keep everything on track, raising ideas and helping out each other. We can also schedule and invite people to join a meeting. This is then notified through your calendar, so everyone knows when to not bother you.



Microsoft Teams

3: Home office style

Wake up early. Yes, it may be very tempting to get up at 8:55am (if you start at 9am), walk over and turn on your laptop and start to work. However, you may still be suffering from sleep inertia making you inherently slow. What some of our staff like to do is get up early, say,

at least one hour before, so that your body has time to adjust. You will also have enough time in an hour to get some morning fresh air and make yourself an amazing home cooked breakfast that will fuel you for the day!



Getting dressed when working from home

It's also a good idea to get dressed, not necessarily as formally as when you need to go to the office, but ditch those pyjamas, put on neat clothes and maybe even brush your hair. This helps setting a working day mood

and drawing a line between being at work and being at home, and especially, staying ready if you need to show up on a video conference with your team or clients.

4: Structure the day as you would in the office

When working from home, you are your own personal manager. You may see that work may creep into your personal time. Set clear times for when you are working and when you are not and stick to them. This will help curb burnout and will allow you to have personal time and rest before the next workday. Write down clear guidelines for each individual task and finish each step on time to meet the normal deadline as usual. The best practice here is to create an automatic time-tracking model or application, which allows you to check whether you are sticking to the schedule.

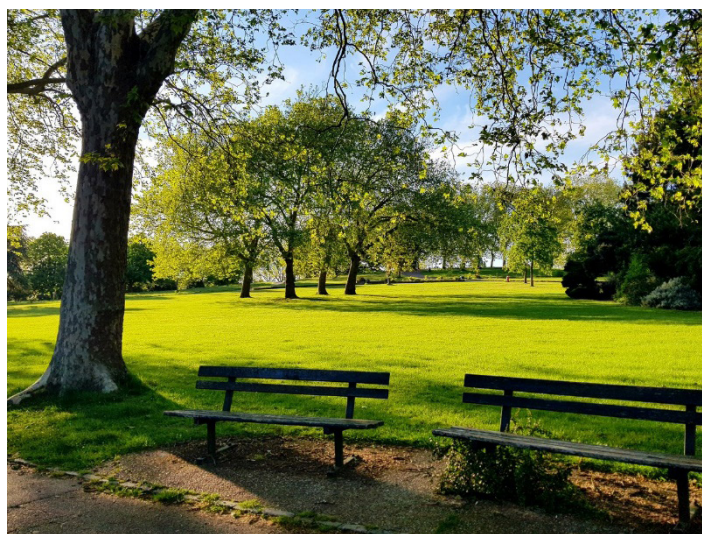


Nine to five

More importantly, working from home may get us mixed up between work life and home life. If you have other family members or friends living in the same house, they may disturb your work unintentionally. This is especially apparent if you have young children or pets that may be constantly seeking your attention. Although it may be hard to say no to those cute little eyes, you have to set clear boundaries so that you

can get your work done. Also, try to agree with your family members that you are working from home, so you need to avoid doing household tasks during the day. However, we can afford the exception of cooking a quick, fresh lunch, rather than a prepared lunch box, and this is one of the perks of working from home!

5: Ensure your personal wellbeing



Being connected with nature

While ensuring work productivity, mental health and personal wellbeing are also important. Follow the company's policy on break times and take them. Proper rest would guarantee working efficiency. A simple method here is to setup a clock or timer on the screen when you take a break.

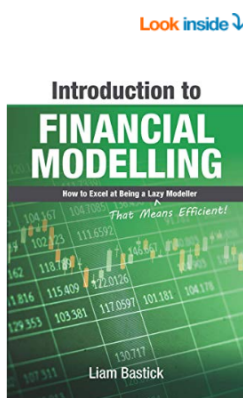
Try to get some exercise into your daily schedule as well. Since all the gyms and indoor sport venues are closed presently, home workouts or sometimes, taking a walk in a green park can keep you stay sane. And always remember to keep to those social distancing rules.

As one who enjoys office life, staying at home all day long, all week long is quite a strange experience. Since we cannot change it, I shall

embrace it. There's likelihood that this may change the future dynamics of business and how people work. Though being a challenge, working from home gives you the independence of managing your own self, saves hours on commuting, and gives time for self-reflection of what is important.

Hopefully, this may help you be successful at home and keep calm. It is important for everyone to do their part now for the good of the population. In the meantime, let's keep our fingers crossed for the storm to pass.

Introduction to Financial Modelling Finally Available on Kindle



Introduction To Financial Modelling: How to Excel at Being a Lazy (That Means Efficient!) Modeller Kindle Edition

by [Liam Bastick](#) (Author) | Format: Kindle Edition

[See all 2 formats and editions](#)

Kindle
\$10.23

Paperback
from \$26.81

[Read with Our Free App](#)

7 Used from \$26.90
12 New from \$26.81

If you have had little formal training in developing financial forecasts in Excel or have ever burnt the midnight oil trying to get a Balance Sheet to balance, then this book is for you. A simple walkthrough of the common perils and pitfalls of financial modelling, this book constructs a solid foundation to build upon (pun most definitely intended). Taking little for granted, Liam examines the common Excel functions and functionalities necessary, emphasises the importance of a standardised and functional layout, explains

[Read more](#)

Length: 328 pages

Word Wise: Enabled

Enhanced Typesetting: Enabled

Page Flip: Enabled

Due to its large file size, this book may take longer to download

It has taken some time, but we got there in the end! So many people have asked about this. SumProduct is proud to announce our first financial modelling book, *An Introduction to Financial Modelling*, is now available on Amazon's Kindle – wherever you may be (physical copies too!).

It even had a good start (albeit in niche sectors!):

Amazon Bestsellers Rank: #7,592 Paid in Kindle Store (See [Top 100 Paid in Kindle Store](#))

#1 in [Computer Networking](#)

#1 in [Business Consolidations & Mergers \(Books\)](#)

#1 in [Business Consolidations & Mergers \(Kindle Store\)](#)

Thanks for bearing with us – and thanks for buying it! And we will have some more news for you later in the year...

Beat the Boredom Challenge

With many of us currently “working from home” / quarantined, there are only so many episodes of *The Ex Factor* you can watch before you start going stir crazy. Some say I reached that point many years ago. To try and keep our readers engaged, we will be reproducing 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...

If you work with Excel in finance, sometimes your data integrity is not as good as you might hope. You may have omissions, duplicates or errors and sometimes, it may contain text. And that last issue can be a real pain in its own right. Let me explain...

This month, we are all going to be bankers and you have been presented with the following report:

Business	Account Type	No of Accounts	Paperwork	Interview	Processing	Postage	Set-up	Overheads
A	Standard	117	3.4	3.0	1.2	2.5	1.5	2.1
A	Deluxe	22	2.1	1.7	3.5	3.8	3.2	1.0
B	Deluxe	49	3.4	1.5	1.5	2.9	3.0	3.3
B	Standard	68	1.8	4.0	3.3	2.4	1.8	1.6

Hourly Rate	\$15	\$24	n/a	\$19	\$29	\$38
-------------	------	------	-----	------	------	------

Total Costs

Business:

Account Type:

Total Costs: <- Your solution goes here

Let me explain the table. Essentially, we have two businesses, A and B, which each have standard and deluxe bank accounts (presumably the latter has money in it!). Cell E17 shows that business A has 117 standard bank accounts for example.

Now here at the bank, we have identified the average time it takes each month to deal with each account (so for instance cell H8 shows each of B's deluxe accounts takes 1.5 hours to process).

Finally, row 11 shows the hourly rates for each process (so cell G11 denotes that interview time is \$24 per hour). Therefore, total costs will be $(117 \times 3.4 \times \$15) + (117 \times 3.0 \times \$24) + (117 \times \dots$

You should note a problem here though. See the lovely shade of red in cell H11? That's because we don't have the data presently and rather than leave the cell blank some kind soul has been “helpful” and typed in “n/a” instead. Lovely.

So here's the challenge. Without adding any helper cells, rows or columns, or using VBA / user-defined functions, can you create a scalable formula which will ignore text and still calculate the total costs associated with business B's deluxe accounts?

For the record, “scalable” means I could add more processes, more types of account and more businesses. Therefore, the following “simplistic” formula

```
=IFERROR((C6=E16)*(D6=E17)*E6*F6*F11,)+IFERROR((C6=E16)*(D6=E17)*E6*G6*G11,)+IFERROR((C6=E16)*(D6=E17)*E6*H6*H11,)+IFERROR((C6=E16)*(D6=E17)*E6*I6*I11,)+IFERROR((C6=E16)*(D6=E17)*E6*J6*J11,)+IFERROR((C6=E16)*(D6=E17)*E6*K6*K11,)+IFERROR((C7=E16)*(D7=E17)*E7*F7*F11,)+IFERROR((C7=E16)*(D7=E17)*E7*G7*G11,)+IFERROR((C7=E16)*(D7=E17)*E7*H7*H11,)+IFERROR((C7=E16)*(D7=E17)*E7*I7*I11,)+IFERROR((C7=E16)*(D7=E17)*E7*J7*J11,)+IFERROR((C7=E16)*(D7=E17)*E7*K7*K11,)+IFERROR((C8=E16)*(D8=E17)*E8*F8*F11,)+IFERROR((C8=E16)*(D8=E17)*E8*G8*G11,)+IFERROR((C8=E16)*(D8=E17)*E8*H8*H11,)+IFERROR((C8=E16)*(D8=E17)*E8*I8*I11,)+IFERROR((C8=E16)*(D8=E17)*E8*J8*J11,)+IFERROR((C8=E16)*(D8=E17)*E8*K8*K11,)+IFERROR((C9=E16)*(D9=E17)*E9*F9*F11,)+IFERROR((C9=E16)*(D9=E17)*E9*G9*G11,)+IFERROR((C9=E16)*(D9=E17)*E9*H9*H11,)+IFERROR((C9=E16)*(D9=E17)*E9*I9*I11,)+IFERROR((C9=E16)*(D9=E17)*E9*J9*J11,)+IFERROR((C9=E16)*(D9=E17)*E9*K9*K11,)
```

might be a solution presently, but (a) won't scale and (b) is absolutely horrible.

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

STOP PRESS: Excel Virtually Global



As the year has progressed and the coronavirus disrupts many plans, SumProduct is getting asked more and more frequently about this year's **Excel Summit South**. We have decided to bite the bullet and accept – for 2020 – it's **not** going to happen.

But we ain't going down without a fight. Instead, we are planning a **MASSIVE** virtual conference with all profits going to a COVID-19 charity. More sessions. More MVPs. Global. It's happening in July and, watch this space, more details shall be coming very soon!

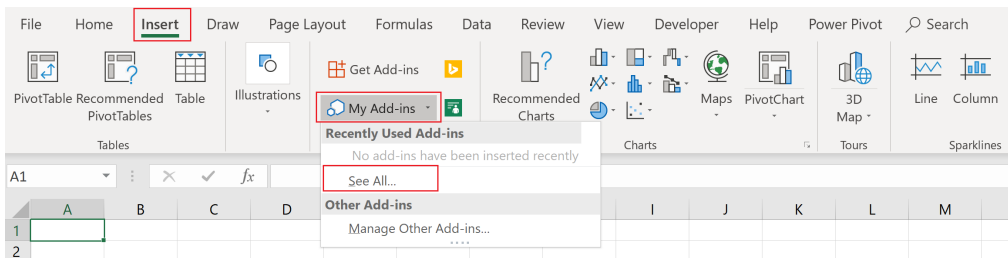
Microsoft Visio Data Visualizer Add-in for Excel

The Microsoft Visio Data Visualizer add-in for Excel is a new way to create data-driven high-quality Visio flowcharts, cross-functional flowcharts and organisation(al) charts – all the shapes, connections and data linking – directly in Excel. The add-in uses Office (Visio) Online to create the

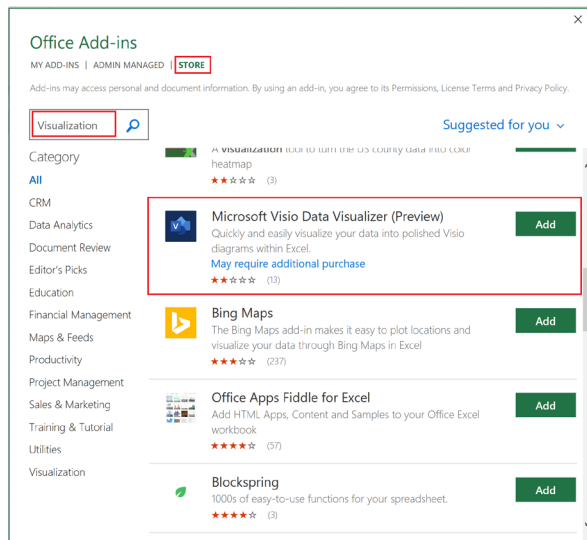
visualisation. The Visio Data Visualizer add-in for Excel is now available in public Preview and will be Generally Available in the near future. You do not need license to use this add-in, however, for additional functionality, such as editing, a subscription is required.

Getting Started

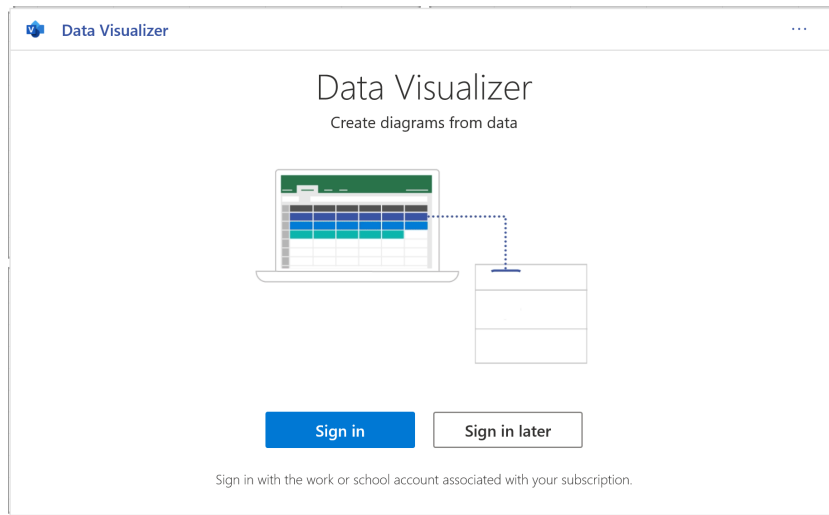
To get this add-in, it's not on the Add-ins tab! Instead, navigate to the Insert tab on the Ribbon, and then from the 'My Add-ins' dropdown list, select 'See All...':



An 'Office Add-ins' dialog will appear. Go to the Store tab and type "Visualization" in the Search box. You should then see 'Microsoft Visio Data Visualizer (Preview)' from the list. Click Add to install the add-in:



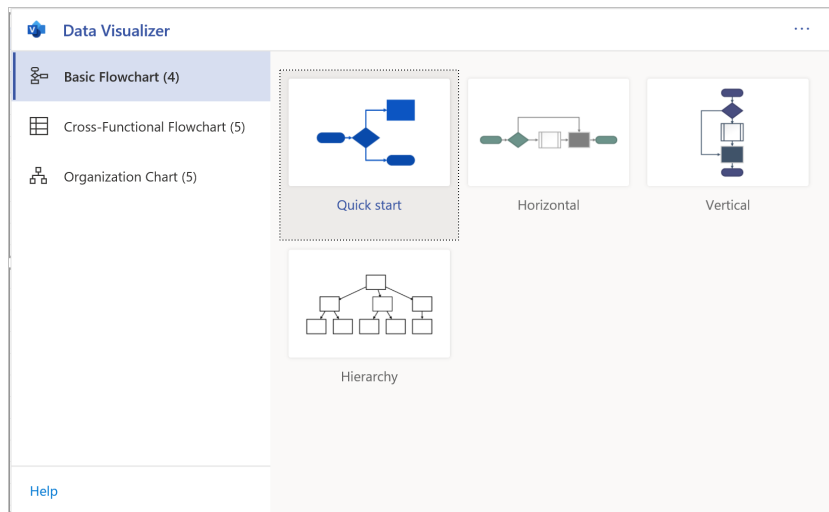
After being installed, the 'Data Visualizer' dialog will be displayed and prompt you to sign in. If you have a work or school account, you can sign in here, or just click 'Sign in later' to skip this step, because the add-in requires no license to use, although Visio subscribers benefit from additional editing capabilities.



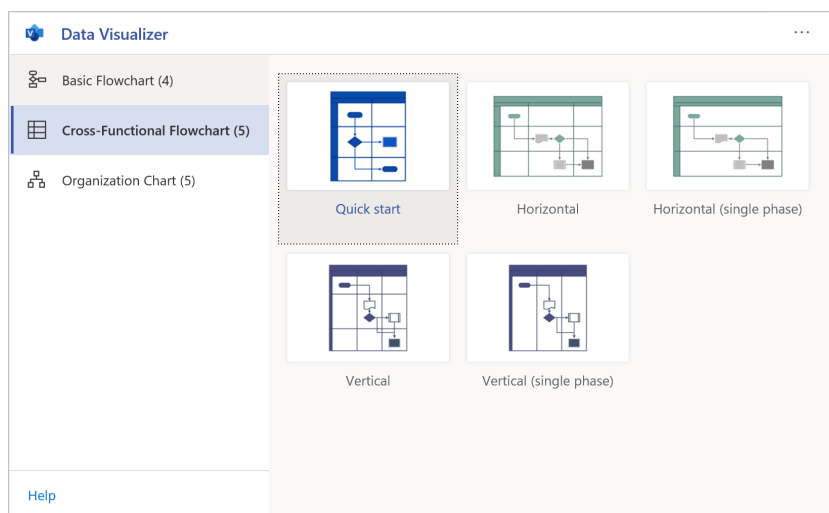
The Data Visualizer add-in comes with three groups of data-linked samples:

- Basic Flowchart
- Cross-Functional Flowchart
- Organization Chart.

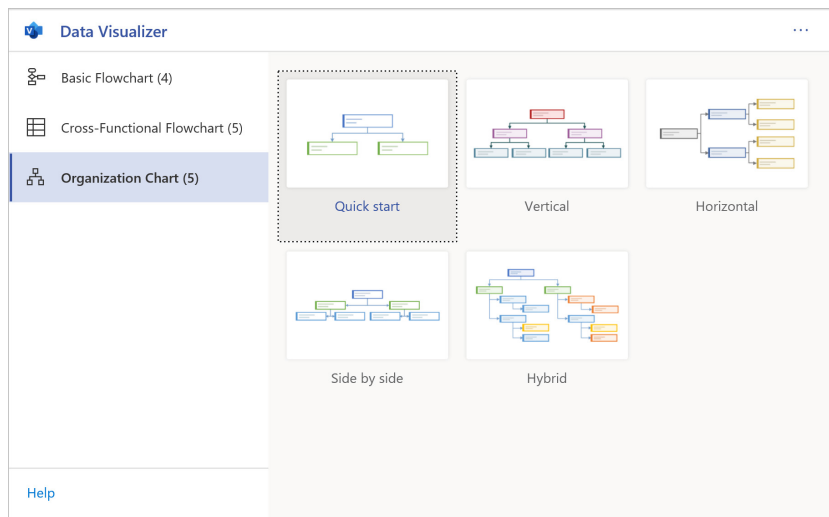
Each diagram category provides a few layout styles and theme options:



Basic Flowchart: Quick start, Horizontal, Vertical, and Hierarchy



Cross-Functional Flowchart: Quick start, Horizontal, Horizontal (single phase), Vertical, and Vertical (single phase)



Organization Chart: Quick start, Vertical, Horizontal, Side by side, and Hybrid

To start, let's choose the 'Quick start' sample from the 'Basic Flowchart' group. The add-in will create a sample diagram with an editable sample data table, similar to the one below:

Process Step ID	Process Step Description	Next Step ID	Connector Label	Shape Type	Alt Text
P100	Start	P200		Start	
P200	Decision?	P300,P400	Yes,No	Decision	
P300	Process			Process	
P400	Stop			End	

The Excel workbook must be saved in order to link to the Visio diagram, so let's save the file and click Done. The chart can also be moved by clicking on the chart box and moving it to the desired location:

Process Step ID	Process Step Description	Next Step ID	Connector Label	Shape Type	Alt Text
P100	Start	P200		Start	
P200	Decision?	P300,P400	Yes,No	Decision	
P300	Process			Process	
P400	Stop			End	

Below is a closer look at the data table, where details have been altered to create our customised flowchart. If you hover the mouse over the column headings, you will trigger ScreenTips of what should be done with the highlighted section of the data table, e.g.

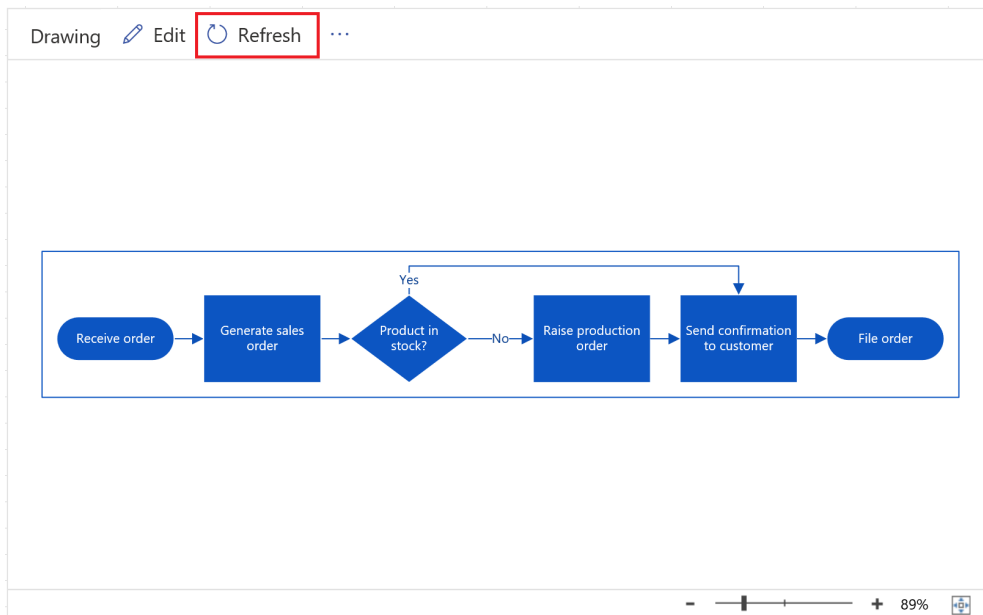
Process Step ID	Process Step Description	Next Step ID	Connector Label	Shape Type	Alt Text
P100	Receive order			Start	
P200	Generate sales order			Process	
P300	Product in stock		Yes, No	Decision	
P400	Send confirmation			Process	
P500	Raise production			Process	
P600	File order			End	

The flowchart contains six steps, each of which has an ID, caption, dependencies, connectors, shape type and alternative text. It is worth noting that particular attention should be paid to the next step ID and connector label fields. In the example below, step P300 – ‘Product in stock?’ has the next step IDs P400 and P500, together with the connector label ‘Yes,No’. This means the ‘Yes’ label is linked to step P400 and the ‘No’ label is linked to step P500.

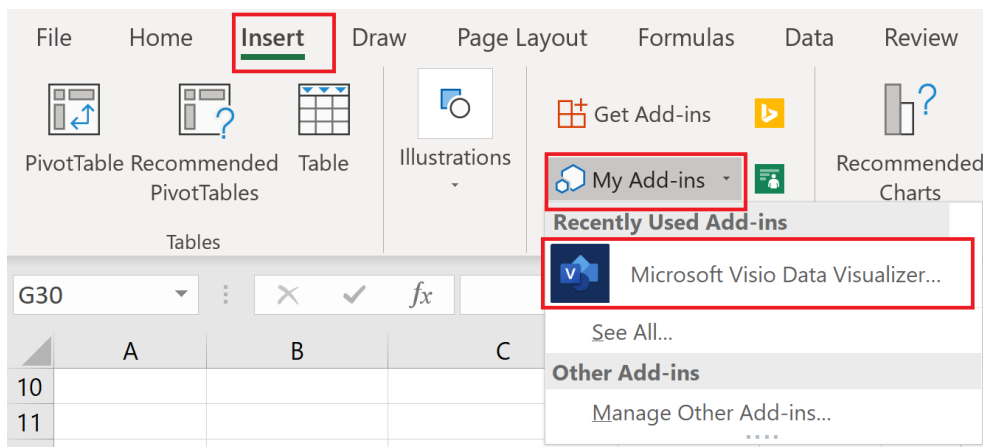
With regard to the shape type, the rounded-corner rectangle for Start/End and diamond-shaped box for Decision, for example, may be defined from the built-in cell dropdown list. There are also custom shapes to choose from, which may be customised should you have a subscription license.

	A	B	C	D	E	F
1	Process Step ID	Process Step Description	Next Step ID	Connector Label	Shape Type	Alt Text
2	P100	Receive order	P200		Start	
3	P200	Generate sales order	P300		Process	
4	P300	Product in stock?	P400,P500	Yes, No	Decision	
5	P400	Send confirmation to customer	P600		Process	
6	P500	Raise production order	P400		Process	
7	P600	File order			End	
8					End	
9					Document	
10					Data	
11					Database	
12					External Data	
13					On-page refere	
14					Off-page refere	
					Custom 1	

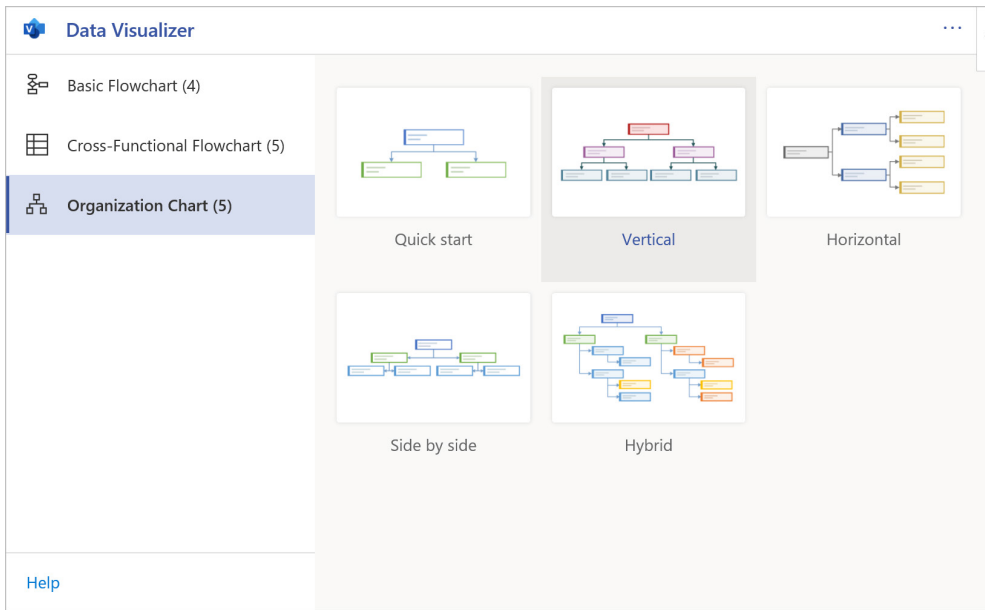
Once editing the data table has been completed, go to the chart box and click Refresh. The flowchart is now created:



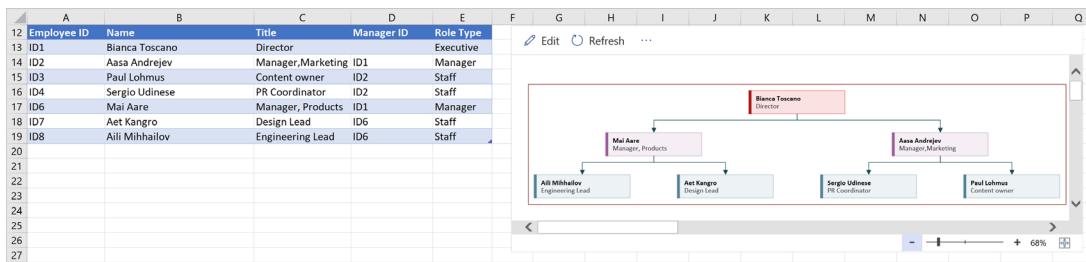
If you want to create another flowchart, you simply need to go back to Insert tab on the Ribbon and choose ‘My Add-ins’ → ‘Microsoft Visio Data Visualizer...’ since this has now been installed:



The category dialog appears, so this time let's create a Vertical organisation chart:

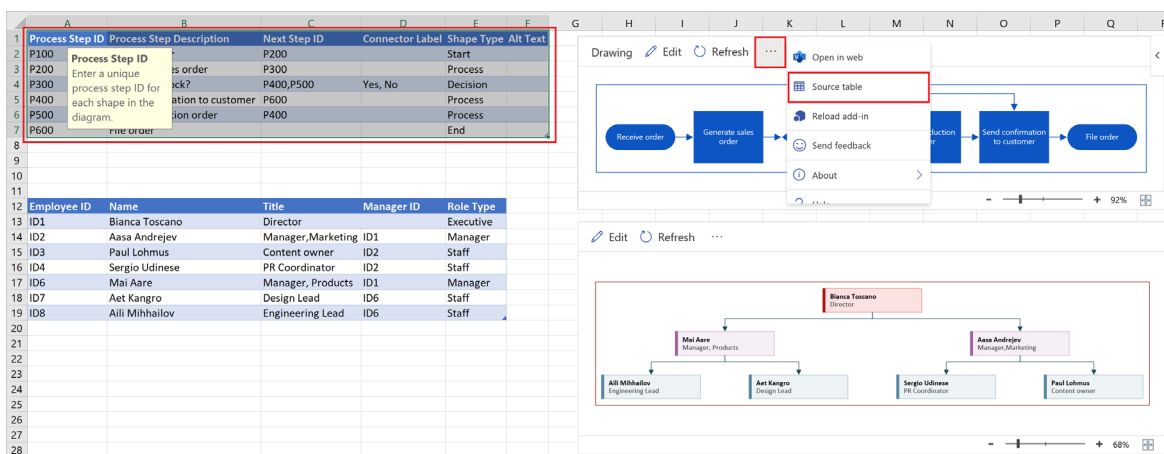


Similarly, the new sample data table and diagram are displayed, where you can change data and refresh the chart box to create the organisation hierarchy:

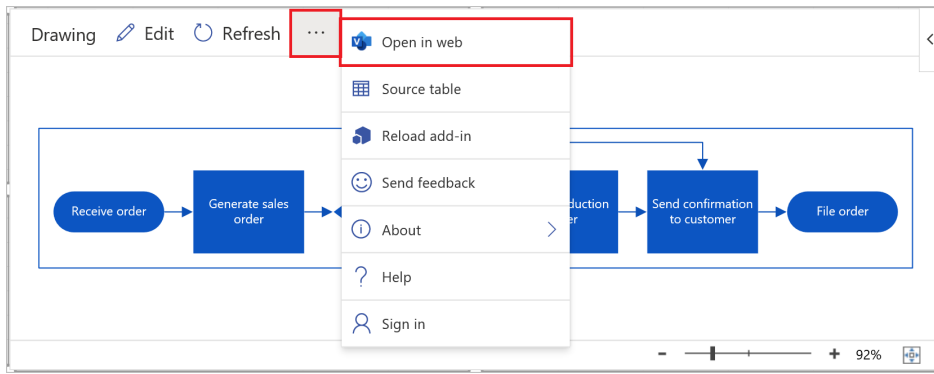


More Actions with Your Diagrams

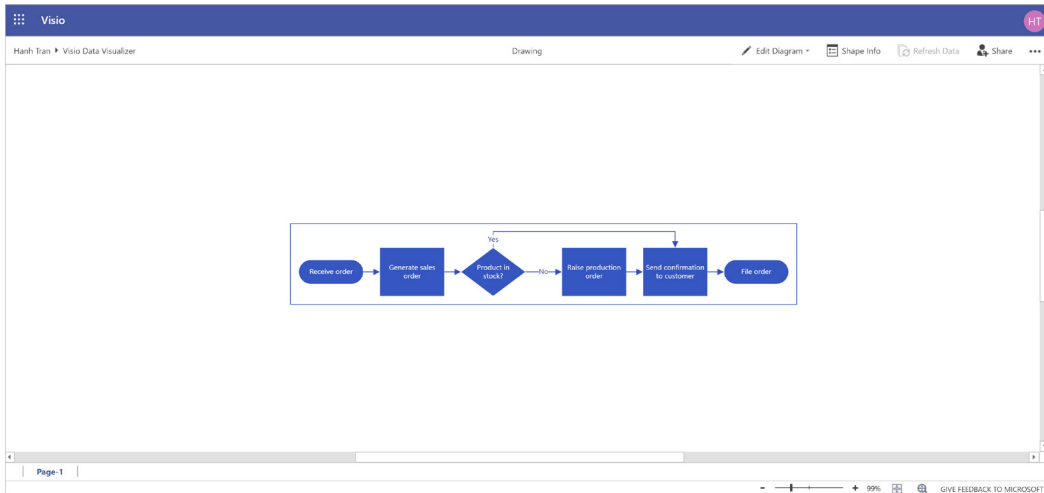
In an Excel sheet with more than one flowchart, to identify the data table linked to the chart, click the ellipses in the chart box and choose 'Source table'. This will cause the linked data table to be highlighted:



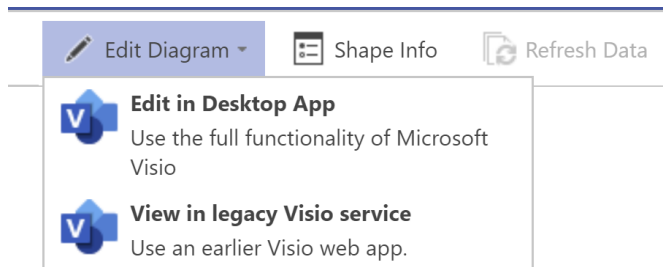
Also, if you are signed in, your Visio diagrams are created as online files and saved in OneDrive or SharePoint by default, so you may view, print and share them with other Office 365 users. To open the diagram in Visio for the web, from the ellipses, choose 'Open in web':



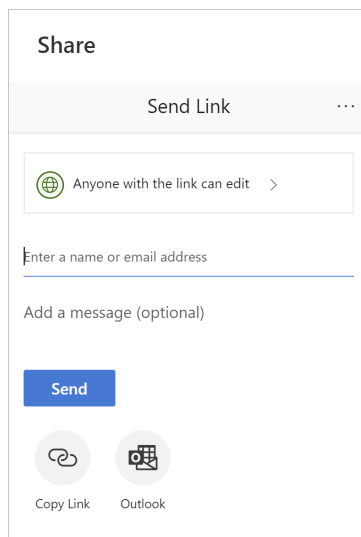
You will be navigated to Visio on the web in your browser, where you can view your flowchart. Options for more actions with the diagram may be found in the top right-hand corner:



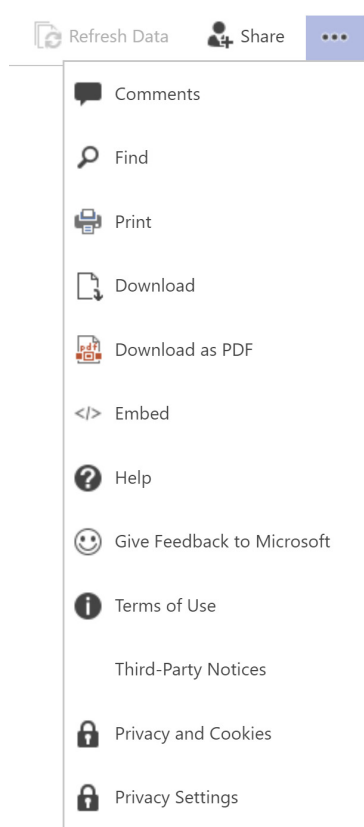
Editing the diagram, such as adding text or images, applying design themes or making other customisation modifications, requires a licence.



Clicking the Share button allows you to share this file with other team members or any Office 365 users:



From the far most right-hand corner ellipse, you have options to add comments, print or download the diagram:



In short, this add-in makes it much straightforward for Excel users to create hierarchy type process maps and flowcharts. From the samples discussed above (but not limited to them), this add-in can be utilised to draw other creative charts, such as decision trees, activity diagrams or conditional probabilities modelling.

Office Scripts Goes Local

We have mentioned Office Scripts previously, and explained it is a new feature for Excel on the web. However, we thought we would revisit this as it is now available for users in what Microsoft calls “country /

region specific data centres”, also known as “Go Local data centres”. This means that although a Preview rolled out in January, certain territories (e.g. Australia, Canada, New Zealand) could not access it – well now you can!

Users with any of the following licences:

- Office 365 Business
- Office 365 Business Premium
- Office 365 Enterprise E3
- Office 365 Enterprise E5

(subject to the name changes – see elsewhere in this newsletter!) may now access Office Scripts in Excel on the web.

This is provided your administrator has opted in. To opt into Office Scripts in the Microsoft 365 Admin Center, if you haven't already:

- sign in to Office 365 with your **work or school account** as a global administrator
- in the Microsoft 365 admin center, choose **Settings** -> **Settings** (you may need to select 'Show all' first)
- select Office Scripts
- check the box for 'Let users automate their tasks in Office on the web' to activate Office Scripts.

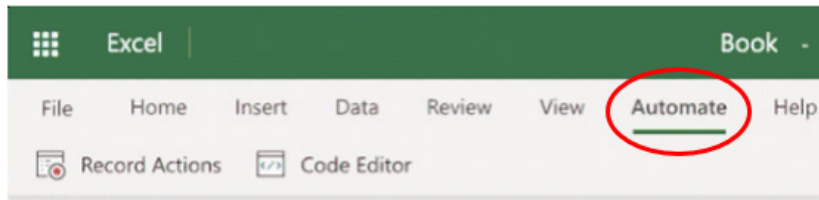
Office Scripts

Office Scripts platform allows users to automate their tasks. This feature is available in preview for Excel on the web. By enabling this feature, users can try Office Scripts and provide feedback while it's in preview.

[Learn more about Office Scripts](#)

Let users automate their tasks in Office on the web

The new feature is incorporated in the Automate tab on the Ribbon for Excel on the web:



This feature allows users to automate some repetitive day-to-day tasks, but in locations where VBA can't. We can record our Excel actions with the Action Recorder and create scripts based on actions. We can also create and edit scripts with the Code Editor. Office Scripts makes the record and replication of Excel actions on different workbooks and worksheets possible, and it may also help you to reduce your whole workflow to a single button press.

The language supporting Office Scripts is JavaScript / TypeScript. Over the past few years Microsoft has been heavily investing in JavaScript as it moves to unify Office across all devices (PC, Mac, Browser, and

Mobile), including automating Excel. With JavaScript, you may theoretically write a single script that would work on any device. Also, JavaScript is extremely fast at pulling data from outside sources, and is especially useful for seamless data integration from different sources. For example, the Google Suite (G Suite) has been using JavaScript as the scripting language, so it is possible to integrate the data stored in Google with API.

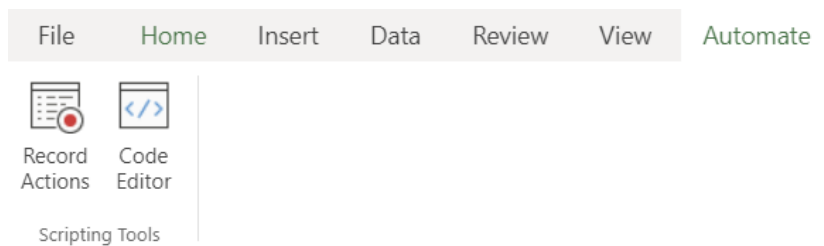
Let's have a look at one simple example to understand how Office Scripts works in Excel on the web. Suppose we have six Sales tables with different business units.

BU1 Customer	BU1 Sales	BU4 Customer	BU4 Sales
A	8	A	10
B	7	B	3
C	5	C	10
D	8	D	6
E	7	E	4

BU2 Customer	BU2 Sales	BU5 Customer	BU5 Sales
A	6	A	9
B	8	B	6
C	1	C	2
D	3	D	2
E	9	E	2

BU3 Customer	BU3 Sales	BU6 Customer	BU6 Sales
A	4	A	5
B	1	B	5
C	9	C	8
D	6	D	2
E	8	E	5

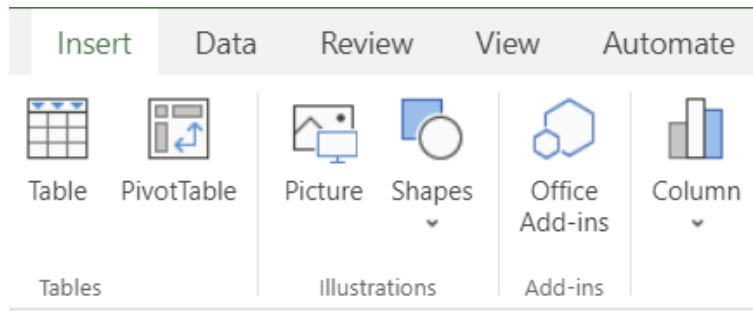
We want to generate charts for each individual business unit Sales table. We can do this by going to the Automate tab and click 'Record Actions'.



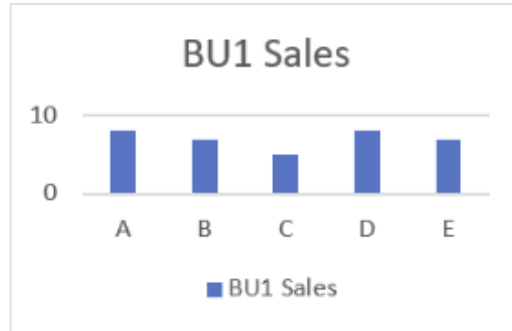
Then, we can start recording our actions in Excel and create our first chart based on the sales data for business unit one (1). In this example, we choose the range **A1:B6**.

	A	B
1	BU1 Customer	BU1 Sales
2	A	8
3	B	7
4	C	5
5	D	8
6	E	7

Click on the Insert tab and choose the Column chart in this case.

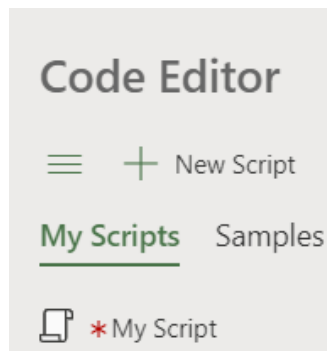


Next, we will have our first simple column chart for business unit one (1) like:



Yes, we can now make the chart prettier by formatting all elements, but that's another example for another day. Here, we simply wish to show you how Office Scripts works. Therefore, at this point, we will stop

recording the code and save the recording with the name 'My Script' for the scripts we created.



In the Code Editor, we can see the code in more detail:

```
Code Editor
Run Save Script
*My Script
1 async function main(context: Excel.RequestContext) {
2   // Insert chart on sheet 'Data'
3   let workbook = context.workbook;
4   let worksheets = workbook.worksheets;
5   let selectedSheet = worksheets.getActiveWorksheet();
6
7   // Insert chart on BU1
8   let chart_1 = selectedSheet.charts.add("ColumnClustered", selectedSheet.
   getRange("A1:B6"));
9   // Resize and move chart Chart 1
10  chart_1.left = 320;
11  chart_1.top = 0;
12  chart_1.width = 170;
13  chart_1.height = 110;
```

In the code above, variables for workbook and active worksheet have been defined. The chart has been added by defining the type of the chart and the range of data source. We may move the chart to its “proper” position by assigning values to different parameters (e.g. **left**, **top**, **width** and **height**).

If we want to create the same chart for the other five (5) business units, we can simply copy and rewrite the code to create more charts based on the existing code and apply changes to the data sources and position of the chart. You could write the code as follows, for example:

```
async function main(context: Excel.RequestContext) {
  // Insert chart on sheet 'Data'
  let workbook = context.workbook;
  let worksheets = workbook.worksheets;
  let selectedSheet = worksheets.getActiveWorksheet();

  // Insert chart on BU1
  let chart_1 = selectedSheet.charts.add("ColumnClustered", selectedSheet.getRange("A1:B6"));
  // Resize and move chart Chart 1
  chart_1.left = 320;
  chart_1.top = 0;
  chart_1.width = 170;
  chart_1.height = 110;

  // Insert chart on BU2
  let chart_2 = selectedSheet.charts.add("ColumnClustered", selectedSheet.getRange("A8:B13"));
  // Resize and move chart Chart 2
  chart_2.left = 320;
  chart_2.top = 120;
  chart_2.width = 170;
  chart_2.height = 110;

  // Insert chart on BU3
  let chart_3 = selectedSheet.charts.add("ColumnClustered", selectedSheet.getRange("A15:B20"));
  // Resize and move chart Chart 3
  chart_3.left = 320;
  chart_3.top = 240;
  chart_3.width = 170;
  chart_3.height = 110;

  // Insert chart on BU4
  let chart_4 = selectedSheet.charts.add("ColumnClustered", selectedSheet.getRange("D1:E6"));
  // Resize and move chart Chart 4
  chart_4.left = 500;
  chart_4.top = 0;
  chart_4.width = 170;
  chart_4.height = 110;

  // Insert chart on BU5
  let chart_5 = selectedSheet.charts.add("ColumnClustered", selectedSheet.getRange("D8:E13"));
  // Resize and move chart Chart 5
  chart_5.left = 500;
  chart_5.top = 120;
  chart_5.width = 170;
  chart_5.height = 110;

  // Insert chart on BU6
  let chart_6 = selectedSheet.charts.add("ColumnClustered", selectedSheet.getRange("D15:E20"));
  // Resize and move chart Chart 6
  chart_6.left = 500;
  chart_6.top = 240;
  chart_6.width = 170;
  chart_6.height = 110;
}
```

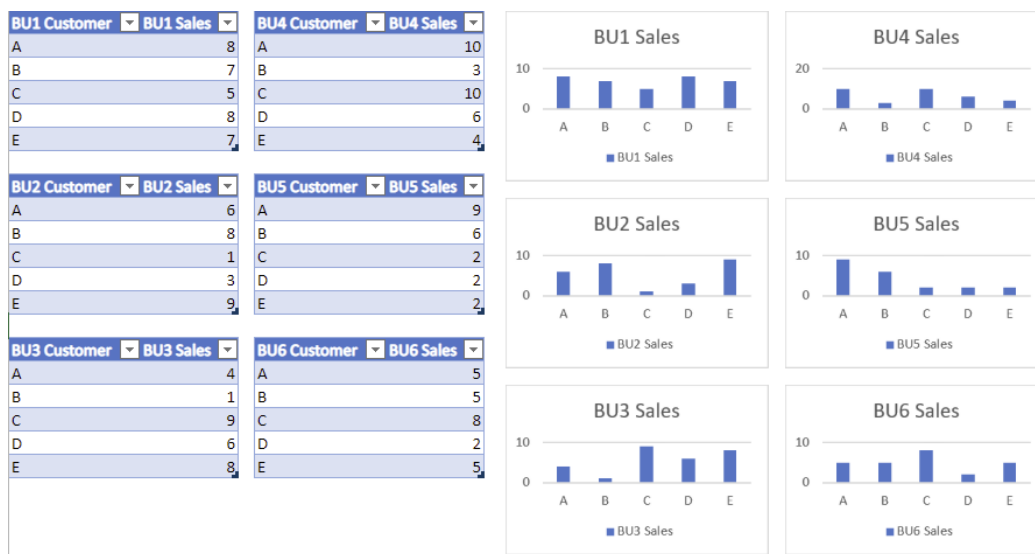
In this routine, we repeat the steps in creating a column chart for each business unit and make adjustments to the parameters of that chart’s position. Then, we click the ‘Run’ button in the Code Editor.

```

Code Editor
Run Save Script
*My Script
1  async function main(context: Excel.RequestContext) {
2    // Insert chart on sheet 'Data'
3    let workbook = context.workbook;
4    let worksheets = workbook.worksheets;
5    let selectedSheet = worksheets.getActiveWorksheet();
6
7    // Insert chart on BU1
8    let chart_1 = selectedSheet.charts.add("ColumnClustered", selectedSheet.
   getRange("A1:B6"));
9    // Resize and move chart Chart 1
10   chart_1.left = 320;
11   chart_1.top = 0;
12   chart_1.width = 170;
13   chart_1.height = 110;

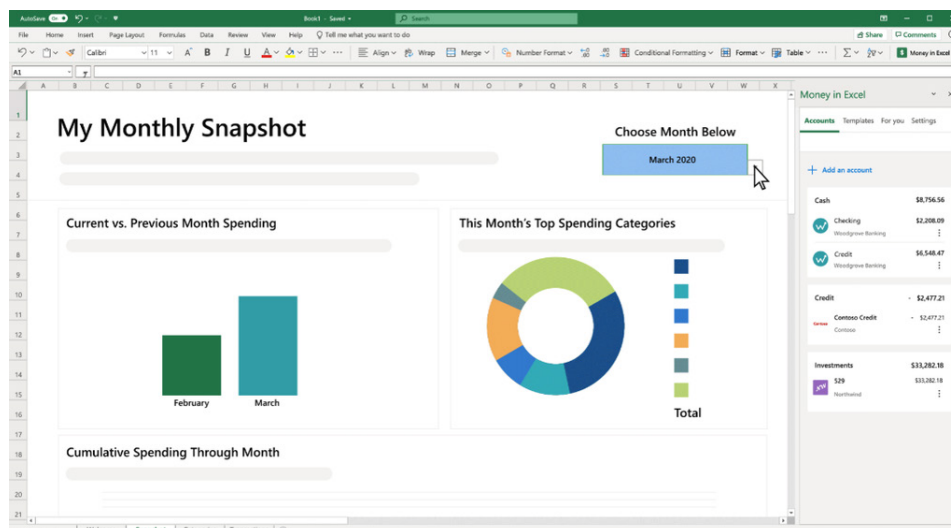
```

The result dashboard would be like:



We can update the Sales data on the left side of the dashboard and open the Code Editor to run the scripts to update the charts on the right side automatically.

Free Excel Money



Well there's a headline! In the current environment, we think it's fair to say there are a few people wanting to keep tabs on their spending and searching for free templates.

Microsoft has announced there is now Money in Excel (alas, not literally). Aimed chiefly at personal users, it's provided to Microsoft 365 subscribers. It is a new feature that will allow you to manage, track and analyse your money and spending in Excel, using financial summaries and data visualisations.

This feature will allow you to connect your bank and credit card accounts to Money in Excel, so you can import transactions and account balances

automatically and create a personalised summary workbook (you might wish to password protect this file!).

Money in Excel can help you recognise and improve your spending habits by providing insights on your spending and proactive alerts about price changes for recurring payments, bank fees, overdraft warnings, and more. It would be good if it will help me - every time I try and calculate a summary of my debt I get the 'Not Enough Resources to Display' error...

Money in Excel will start to become available in the coming months in the US first.

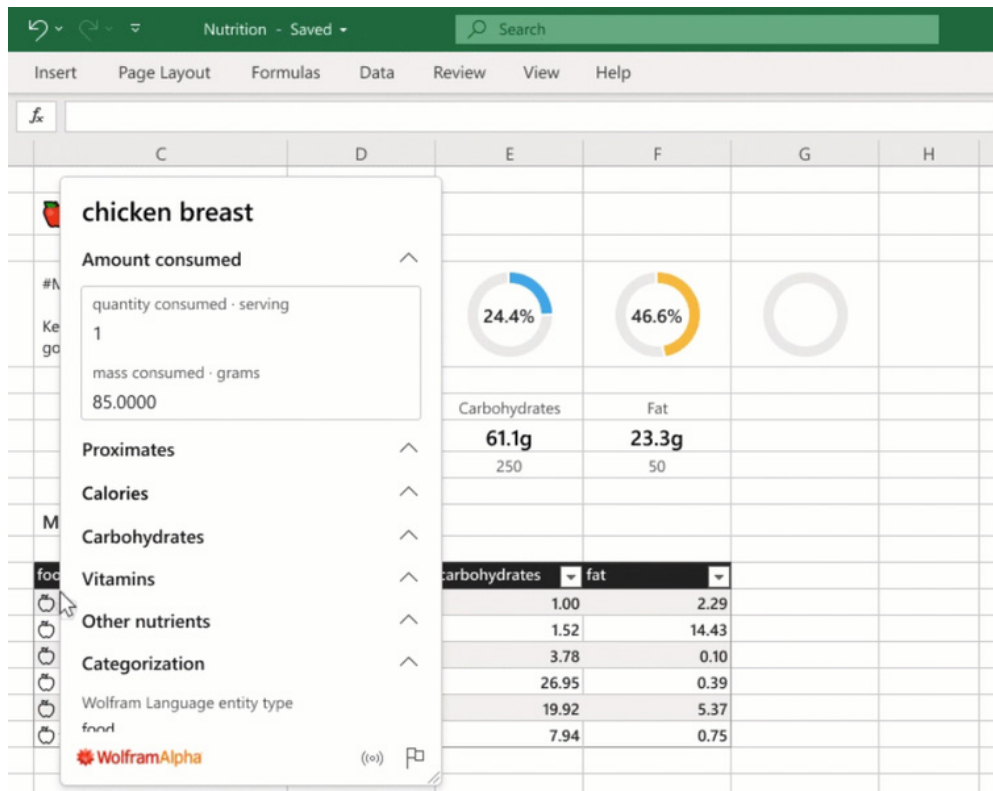
New Data Types and Smart Templates Coming to Excel

Let's be honest: we all use Excel for many different things, business and personal, simple to complex. Power Query and Power Pivot have helped us to find, aggregate, organise and refresh our data in Excel, but it can still take a while to construct.

But it's getting simpler every day.

Microsoft has recently announced another way to interact with your data. New data types are coming. With these new data types, it will become even easier to turn your collated data into digestible information as Excel provides deeper insight for more than 100 topics. These include food, movies, places, chemistry, and even Pokémon (come on everyone,

get a life!). All you have to do to save time is convert your plain text and numbers into a new data type and Excel will surface both rich visual and interactive data cards and images that provide a better representation of your data.



Microsoft has provided an example:

Say you want to start keeping better track of your nutrition. Simply enter a food like "avocado" and by converting it to a Food data type, Excel will help you access its nutritional information. If you are vegan, maybe your family is considering adopting a dog (no, we are not suggesting you eat a dog). You can evaluate different types of breeds using the Animal data type, which provides you with images, facts and their temperaments, so you can easily create a table to compare them.

With data types, the associated information may be easily updated, so you always have the latest and relevant data. Right now, anyone may access the Stocks and Geography data types in Excel for the web. However, Microsoft 365 subscribers will have exclusive access to the

over 100 new data types powered by Wolfram Alpha. As a Microsoft 365 subscriber, data types will give you the opportunity to:

- Easily gather, organise and analyse the information you need in one place without the hassle of finding, copying and refreshing the data
- Use new smart templates with data types to do more. They will help you prepare to move to a new city, track your nutrition, etc.

The new data types and smart templates will be available to Office Insiders shortly and to Microsoft 365 Personal and Family subscribers in English in the coming months. Watch this space!

Office 365 Renamed

Just when you thought you had got your head around all the different versions of Microsoft Office, someone in their Marketing team thought it would be a fantastic idea to rename the products. Let's be honest, you wouldn't want anyone to buy the correct version for their organisation and / or family, now would you?

Here's the official name changes:

Office 365 Business Essentials	is now	Microsoft 365 Business Basic
Office 365 Business Premium	is now	Microsoft 365 Business Standard
Microsoft 365 Business	is now	Microsoft 365 Business Premium
Office 365 Business	is now	Microsoft 365 Apps for business
Office 365 ProPlus	is now	Microsoft 365 Apps for enterprise

Well, that's not confusing: I can imagine some will think their products have been downgraded.

The names updated automatically from 21 April (National Kindergarten Day – ah yes, that makes sense...). Please don't shoot us messengers.

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 look at declaring variables.



One of the most useful things you can do in VBA is to store values in the program's memory, so that you can use it in the future. In Excel, we would normally do this in cells – put a value in a cell, then refer back to that cell when you want to use it. In VBA, we do this with **variables**.

Variables can be used to store any type of object or value. For example,

you could create a variable to store a particular sheet or workbook. You can create a variable to hold a number, or even an array of numbers. You can use variables to store text. However, you generally want to define what each variable will store in advance. We can do this by dimensioning (which is really a fancy way for telling the computer to allocate space for) variables:

```
Dim <VariableName> as <VariableType>
```

For example, we can use the following to create a variable for our message box:

```
Dim MessageBoxText as String
```

A String is the variable type. This tells Excel that the value being stored in this variable is text, rather than numeric or some other form of variable.

We can then add a value to this variable, using the code:

```
<VariableName> = <Value>
```

Again, in the case of our message box, we can use:

```
MessageBoxText = "Hello World!"
```

This effectively tells Excel that we want to find the variable called "MessageBoxText" and set it to represent the value "Hello World!". We need to note that the type of variable is important in defining how the variable will take values. If we set the variable to be equal to "3", then it will take on the value, but it will treat it as a String type, meaning that it can't be used in some types of mathematical operations that do not automatically convert text to numbers.

In some cases, we might need to use:

```
Set StartingWorkbook = ActiveWorkbook
```

Let's take a look at some other examples of variables:

- Dim StoredValue as Integer
(set up a variable that contains integers, i.e. whole numbers)
- Dim DetailedValue as Double
(set up a variable that contains decimal numbers)
- Dim FlagValue as Boolean
(set up a variable that contains True and False values)
- Dim StartingWorkBook as Workbook
(set up a variable that contains a reference to the workbook, so that you can use that in other code elements, e.g. StartingWorkbook.Sheets("Sheet1")).

More next time!

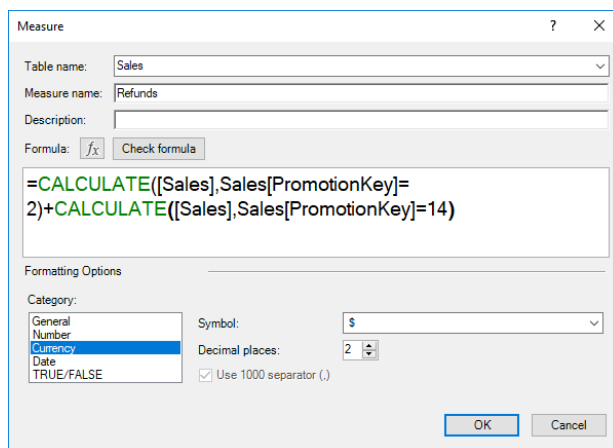
Power Pivot Principles

We continue our series on the Excel COM add-in, Power Pivot. This month, we continue our discussion on CALCULATE.

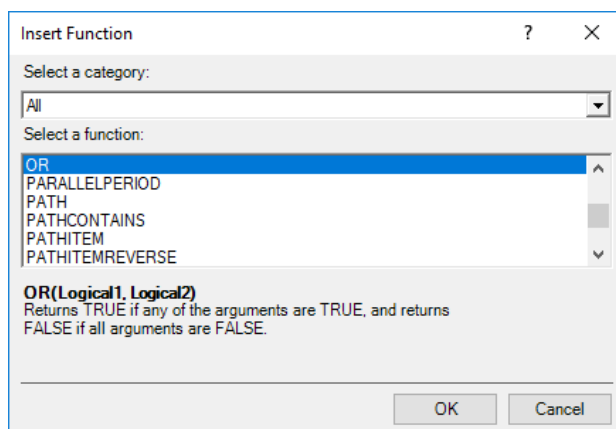
As we explained last month, in Power Pivot, we may use the **CALCULATE** function to filter our total sales amount by other fields, such as **Product Type**, **Weeks** or **Months**. Last month's article just covered a single criterion, so let's extend it: what if we have two criteria that we wish to filter by instead?

Imagine we have data which includes four **PromotionKey** types: 1, 2, 13 and 14. Promotion keys 2 and 14 represent refunds. The aim is to ascertain the total amount of refunds.

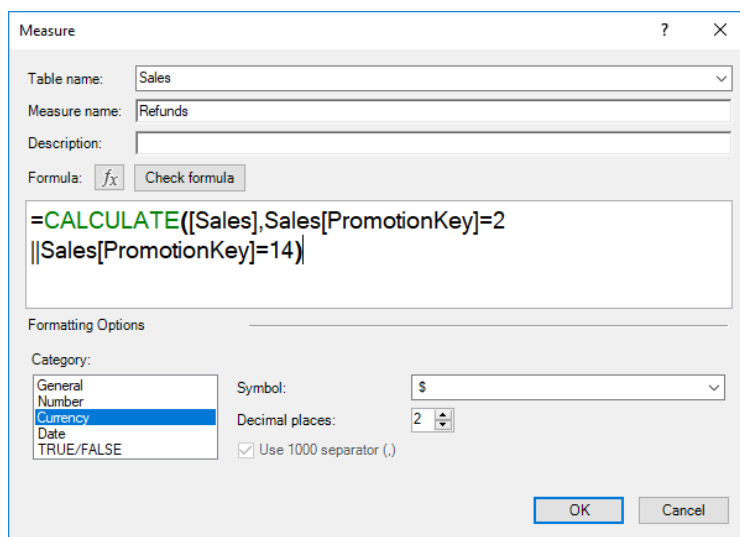
The solution would be simple:



We simply add two formulae together. However, in another subtly different scenario, what if we only want to sum the refunds if it is classified as promotion key 2 or 14? We could use the **OR** function:



This works like the Excel version of the function. However, there is an alternative which has no Excel equivalent – the **OR** operator:



The **OR** operator is two straight lines '||', (you get this by holding down the **SHIFT** key and pressing the Backslash '\ key twice). Operators tend to calculate faster than functions, so with large amounts of data this may be an important consideration.

Further, the **OR** function in DAX accepts only two (2) arguments. If you need to perform an **OR** operation on multiple expressions, you can create a series of calculations or, better, use the **OR** operator (||) to join all of them in a simpler expression.

More *Power Pivot Principles* next month.

Power Query Pointers

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

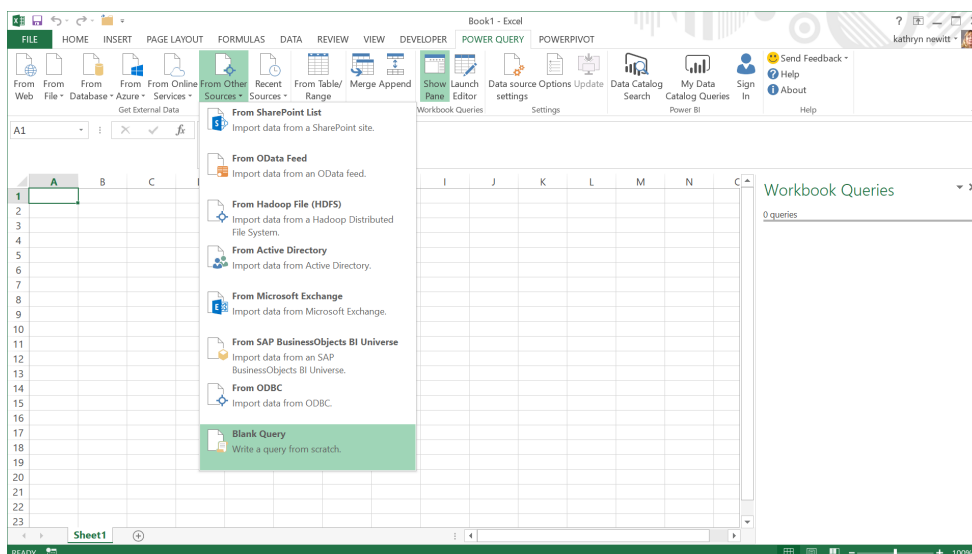
We all like finding ways to save time. That's why I get someone else to write this. Therefore, finding a method to create data quickly is very useful. This particular method of creating data is ideal for those

scenarios where you don't want Excel users to change any values. Since the values are specified in **M** code, the only way to change them is to change the query.

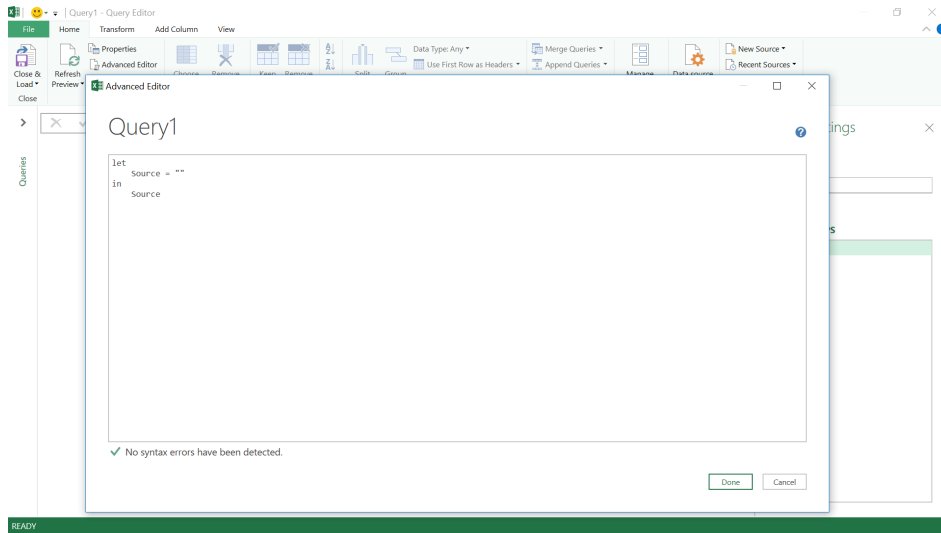
The Microsoft description of the **#table()** function is shown below:

- **#table(columns as any, rows as any)** as any
- Creates a table value from columns (**columns**) and the list (**rows**) where each element of the list is an inner list that contains the column values for a single row; **columns** may be a list of column names, a table type, a number of columns or null.

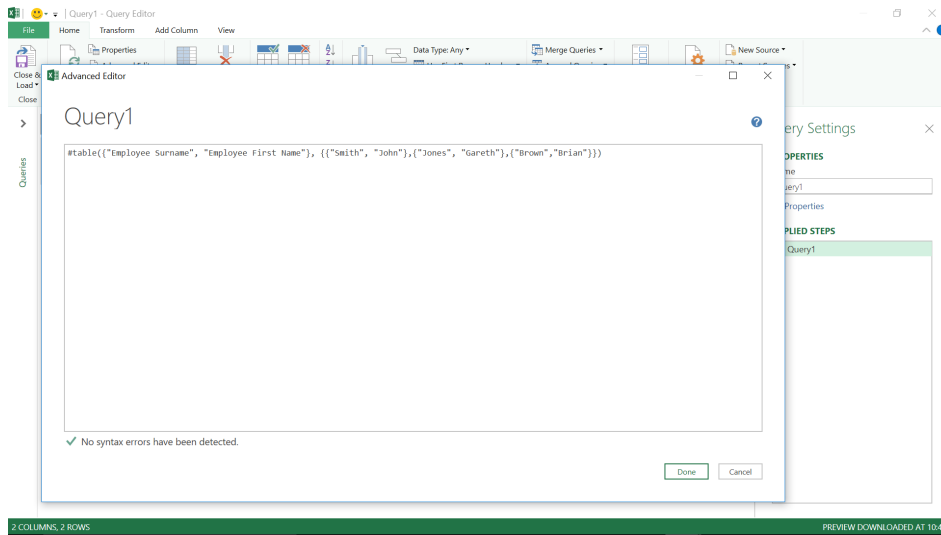
This all sound very simple...or not. What I need is an example. In an empty workbook, elect to create a new, empty query (Blank Query):



In the Query Editor, choose the 'Advanced Editor' option from the Home tab.



Let's begin with something simple: the aim is to create a small table with some basic employee data, none of which is genuine...

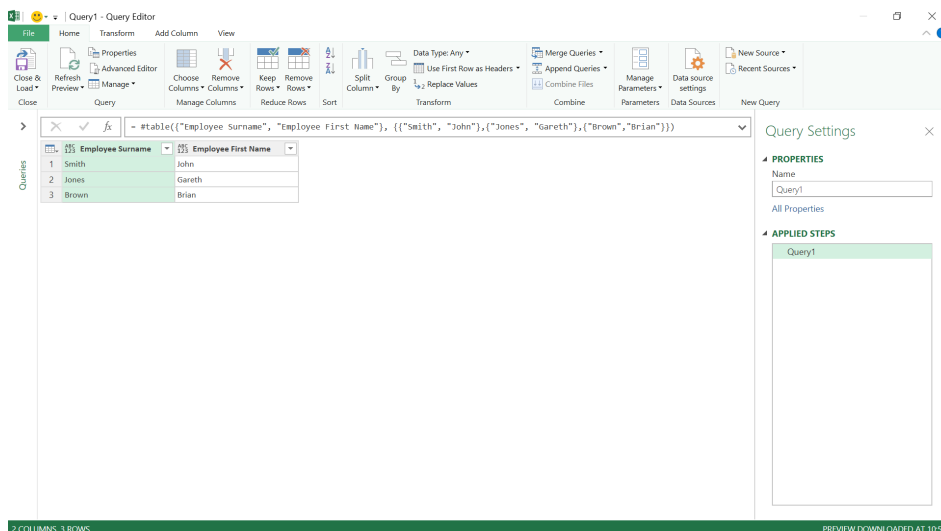


The M code is as follows:

#table({"Employee Surname", "Employee First Name"}, {"Smith", "John"}, {"Jones", "Gareth"}, {"Brown", "Brian"}))

This means, create a table with two columns (these are in a list), **Employee Surname** and **Employee First Name**. Then, the values for each pair of columns are provided three times. Each set of data (surname value and first name value) is grouped into a list and all three pairs are

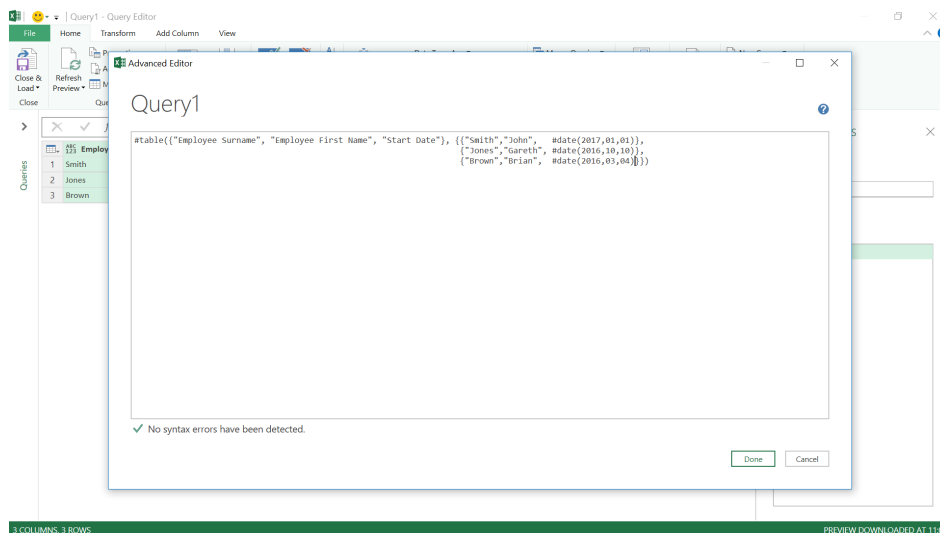
grouped into a larger list. The list is indicated in each case by the use of curly brackets { }. This makes the 'inner list' in the description a little clearer, as it just means that I am specifying a list of values for each row, and the rows belong to a list too. Then, select 'Done' to create the table.



The columns in the example above use text values, but you may use other data types in my columns; it depends what you wish to create. We will decide to add a start date for our three employees. For this we will use another function, **#date**:

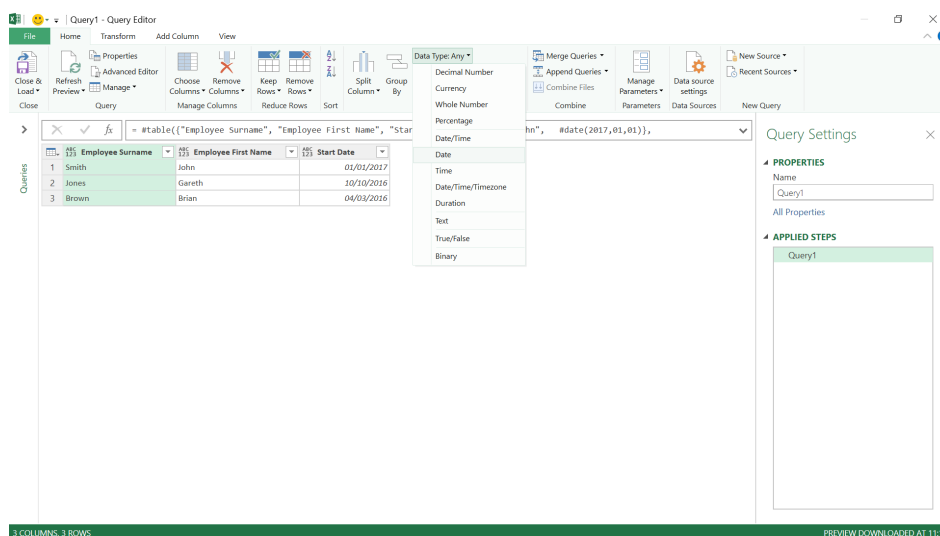
- **#date**(year as number, month as number, day as number) as date
- Creates a date value from year (**year**), month (**month**), and day (**day**). This raises an error if the following conditions are not true:
 - o $1 \leq \text{year} \leq 9999$
 - o $1 \leq \text{month} \leq 12$
 - o $1 \leq \text{day} \leq 31$.

This function is easy to follow. Therefore, let's go back into the Advanced Editor and expand the lists, viz.



As the line is getting longer, we will organise my **M** code so that we may follow where the lists are constructed to make it easier to add to our table. (we have covered basic **M** code syntax and how to make it easier to read in earlier articles)

Once we are happy with what has been added, click 'Done'.



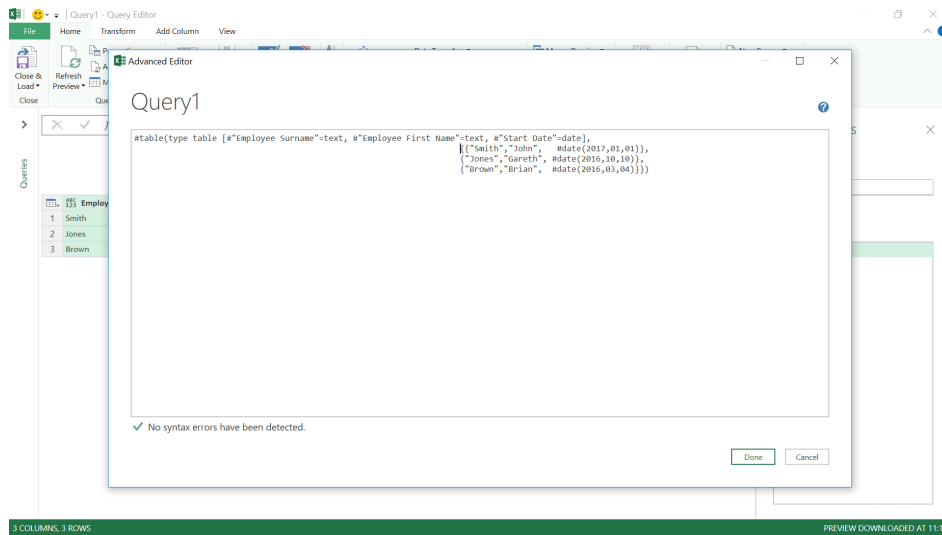
Power Query has added the column as type 'Any', so we should opt to change it to a date. However, this would be better if it were to happen when we create the table, rather than adding another step. There is a way to specify the data type. The syntax is slightly different, as the method of defining the table changes. Instead of using a list of column names,

#table({"Employee Surname", "Employee First Name", "Start Date"})

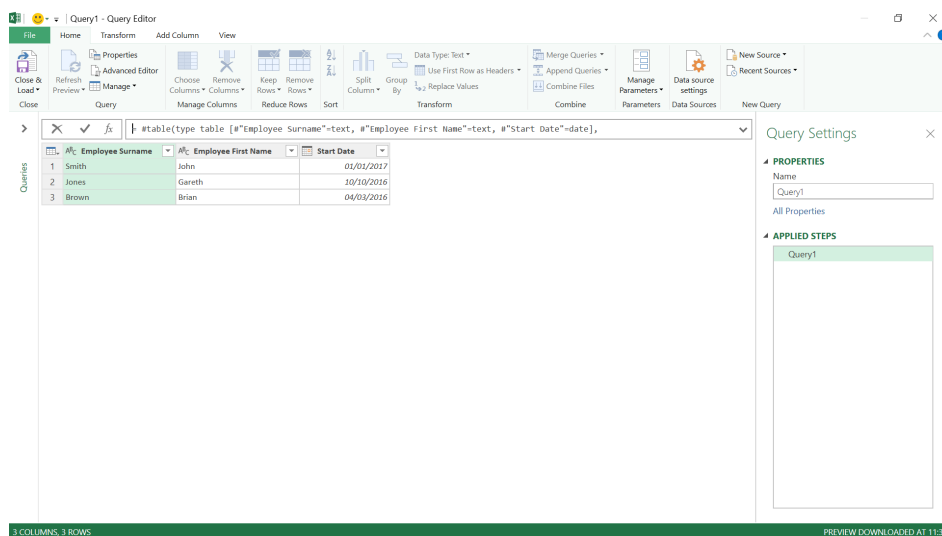
specify the name and type for each column (and the table type) as follows:

#table(type table [#"Employee Surname"=text, #"Employee First Name"=text, #"Start Date"=date])

Now, we may go back to the Advanced Editor and amend our code:



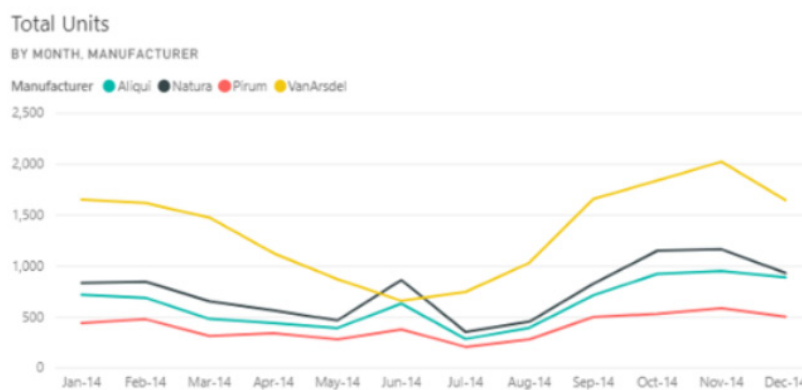
Choose 'Done' to make the changes.



All the columns in the newly created table have been given the correct type without the need for a 'Change Type' step. Easy when you know how... More next month!

Power BI: Copy a Visual as an Image

Whilst we wait for the latest updates to be announced, we are aware that there is now the ability in Power BI to copy a visual as an image to your clipboard. With just one click, you can easily share a captured image of a dashboard tile or a report visual, e.g.



[Open in Power BI](#)

Total Units by Month and Manufacturer

Confidential info

Data as of 3/5/2020, 3:14:42 PM

Filtered by **Manufacturer** (is Aliqui or is Natura or is Pirum or is VanArsdel), **Year** (is 2014)

You can now copy a static image of the visual in the Power BI Service along with the visual's metadata including:

- a link back into the Power BI dashboard or report
- the title of the report or dashboard the content was copied from
- whether the image contains confidential information
- the time stamp of when the data was last updated
- the filters applied to the visual.

To be clear, this ability is different from 'Copy Visual' in Power BI Desktop, which allows you to copy and paste visuals between report pages while authoring.

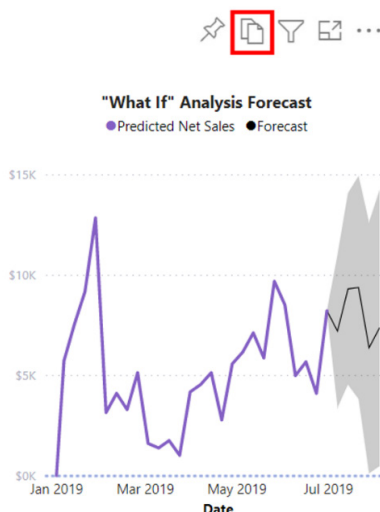
To copy from a dashboard tile:

- navigate to the dashboard you want to copy from
- click on the 'More options' menu on the top right corner of a dashboard tile
- click on 'Copy visual as image'
- when your visual is ready, paste your image using **CTRL + V** or by right-clicking and selecting 'Paste'.



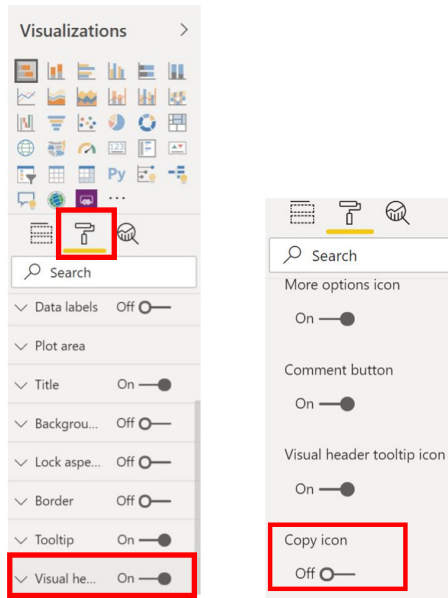
If you want to copy from a Report Visual:

- navigate to the report you want to copy from
- hover over a visual and click on the 'Copy visual as image' icon on the visual header
- when your visual is ready, paste your image using **CTRL + V** or by right-clicking and selecting 'Paste'.

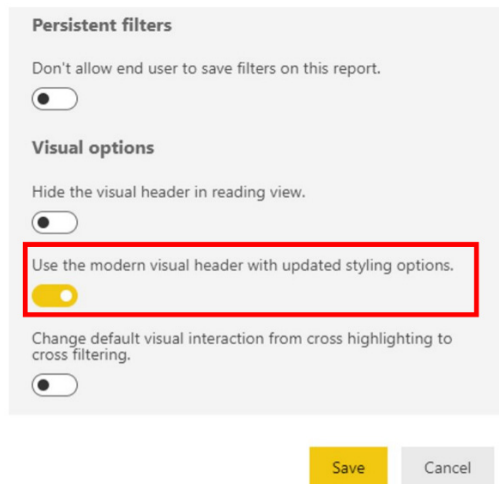


To toggle 'Copy Visual' off for individual report visuals:

- in Edit mode, navigate to the 'Visual Formatting' pane
- under 'Visual Header', toggle off the 'Copy icon'
- save changes, and the Copy icon won't appear on that report visual.

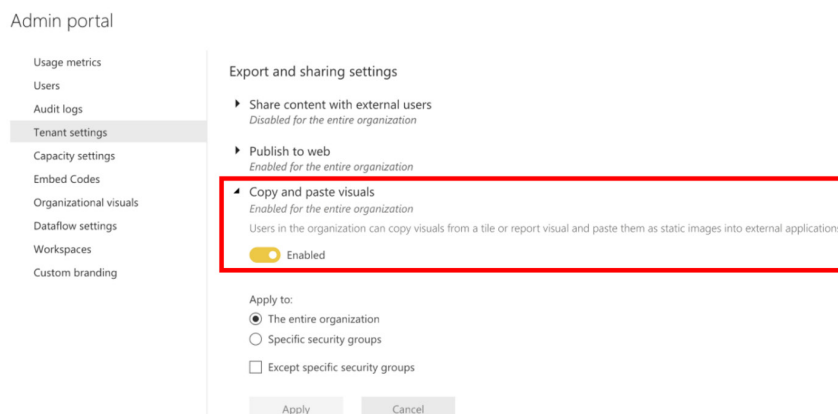


If you are unable to find the 'Visual Header' setting, turn on the modern visual header option under 'Report Settings'.

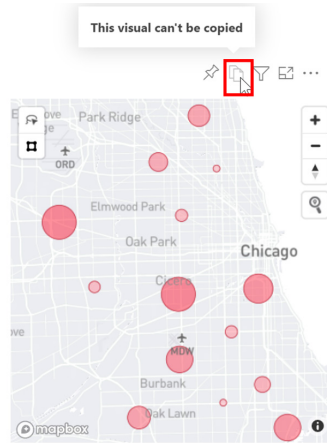


Administrators may turn off this feature for specific security groups or the entire organisation through the Admin portal:

- navigate to the Admin Portal
- under 'Tenant settings', find the 'Export and sharing settings'
- under 'Copy and paste visuals', disable this feature for your selected user groups
- save changes, and the specified groups will not be able to copy visuals as an image throughout Power BI.



Sometimes the Copy icon may be disabled on a visual, e.g.

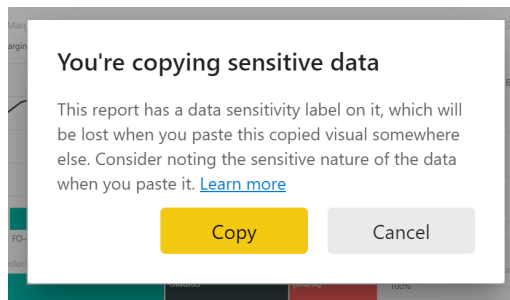


Power BI Service currently supports native Power BI visuals and Certified Visuals. There is limited support for certain visuals including:

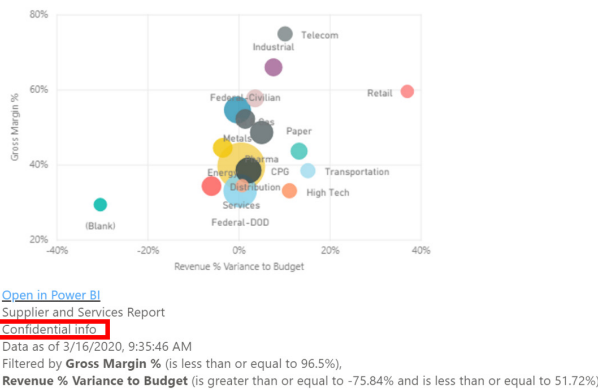
- ESRI and other Map visuals
- Python visuals
- R visuals
- PowerApps
- non-certified custom visuals.

If you want a custom visual to be supported, you will need to certify your custom visual first.

Furthermore, sensitivity labels will not be applied to copied visuals. However, users will be notified that they are taking a screenshot from a sensitive report before copying:



When the content is pasted, there will be information in the metadata letting others know the visual contains confidential data:



There are limitations around 'Copy Visual' including:

- limited support for custom visual formatting such as:
 - o visuals with applied themes and colours
 - o tile scaling when pasting
 - o custom visuals with animations
- copying constraints:
 - o cannot redirect users to content with Odata filters and "sticky" states such as personal bookmarks
 - o pinned live pages are not supported
- applications with limited support for pasting HTML-formatted content from the clipboard may not render everything that was copied from the visual.

But it's a start!

Power BI Service and Mobile Latest Updates

COVID-19 may be slowing down the updates slightly – but only slightly. There have still been quite a few updates for Power BI Service and Mobile:

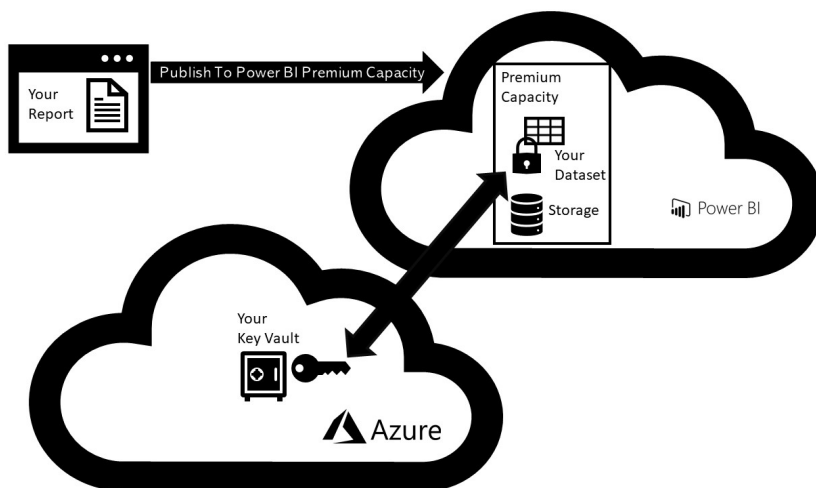
- General Availability of BYOK (Bring Your Own Key) for Power BI Premium
- General Availability of Incremental Refresh with Pro users support
- New Power BI tab for Microsoft Teams
- Featured content on Power BI Home
- Export report to PDF, PPTX and PNG files using Power BI REST API
- Revamped global search in the Power BI Service
- Performance improvements to Multi Geo Premium capacities
- Public Preview of read / write XMLA endpoints in Power BI Premium
- Updates to On-premises data gateway
- Updates to Power BI Mobile apps.

Let's take a quick look...

General Availability of BYOK (Bring Your Own Key) for Power BI Premium

In a recent announcement, Microsoft launched the General Availability (GA) of Bring Your Own Key (BYOK) for Power BI Premium. This feature gives enterprises the ability to configure the encryption key used to encrypt their data when it is stored in the Microsoft cloud. With

BYOK, each premium capacity can be set to encrypt data at rest using separate keys. As a result, you can exercise control over each capacity separately, revoke your organisation's keys and make the data in each separate capacity unreadable to the service within one hour.



General Availability of Incremental Refresh with Pro users support

With the announcement of Incremental Refresh reaching General Availability (GA), Microsoft went a step further and also made it available in Power BI Pro. This is good news. Incremental Refresh is a critical capability that unlocks actionable insights for intelligent decision-making over very large datasets. This illustrates how Power BI provides a simplified management experience for enterprise BI. Where traditionally (e.g. on Azure Analysis Services) customers would write and maintain complex code for this purpose, the Incremental Refresh feature allows the definition of a refresh policy using a simple dialog. The Power BI Service then does the work of managing partitions for optimised data loads. Together with the large models feature, Incremental Refresh can enable datasets to grow to very large sizes. Refreshes are faster and more reliable, so that you don't need to reload all the historical data every time.

Incremental refresh

You can improve the speed of refresh for large tables by using incremental refresh. This setting will apply once you've published a report to the Power BI service.

① Once you've deployed this table to the Power BI service, you won't be able to download it back to Power BI Desktop. [Learn more](#)

Table	Incremental refresh
FactInternetSales	<input checked="" type="checkbox"/> On

Store rows where column "OrderDate" is in the last:
5 Years

Refresh rows where column "OrderDate" is in the last:
10 Days

Detect data changes [Learn more](#)

Only refresh complete days [Learn more](#)

[Apply all](#) [Cancel](#)

New Power BI tab for Microsoft Teams

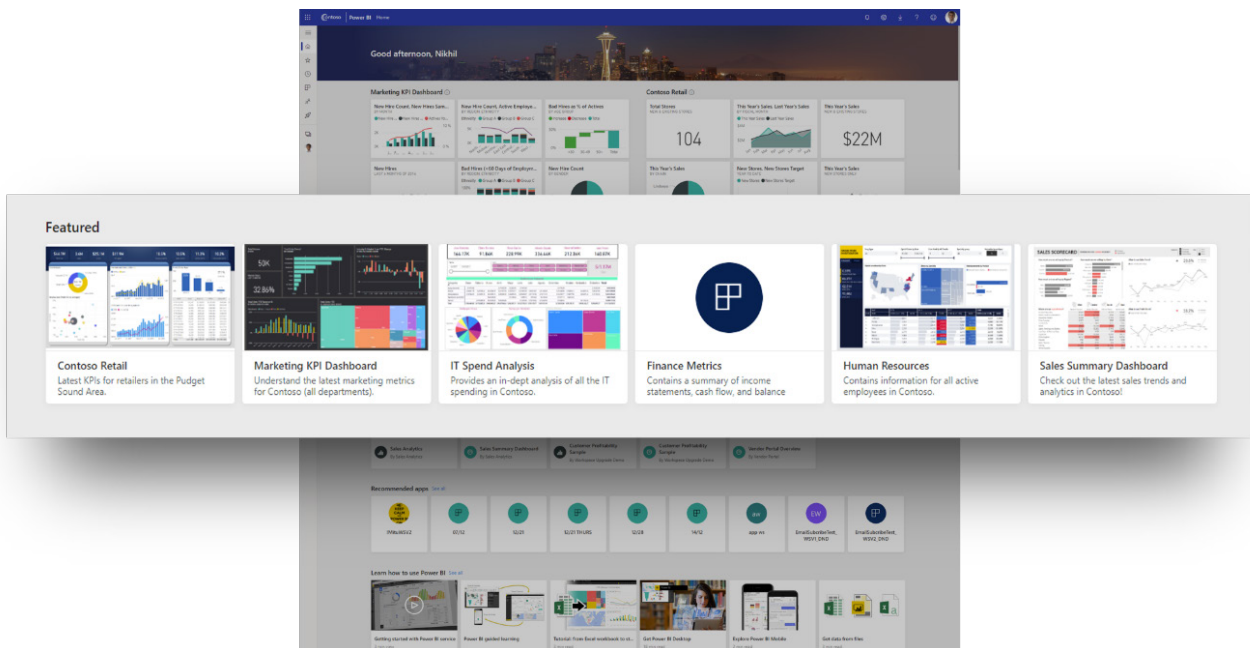
A new Power BI tab has been added for Microsoft Teams. Now you can show graphically in real-time how long you are all waffling for. The new tab supports reports, which are either found in the new workspace experiences, Power BI apps or simply created as a paginated report.



Featured content on Power BI Home

It's now easier for organisations to set the default recommended content for their end users by introducing Featured content on Power BI Home. With this feature, administrators or designated promoters will have a dedicated section on Home to promote reports, dashboards and

applications. By setting default content, organisations can ensure new users will discover relevant content and have a standardised experience for navigating within the Service.



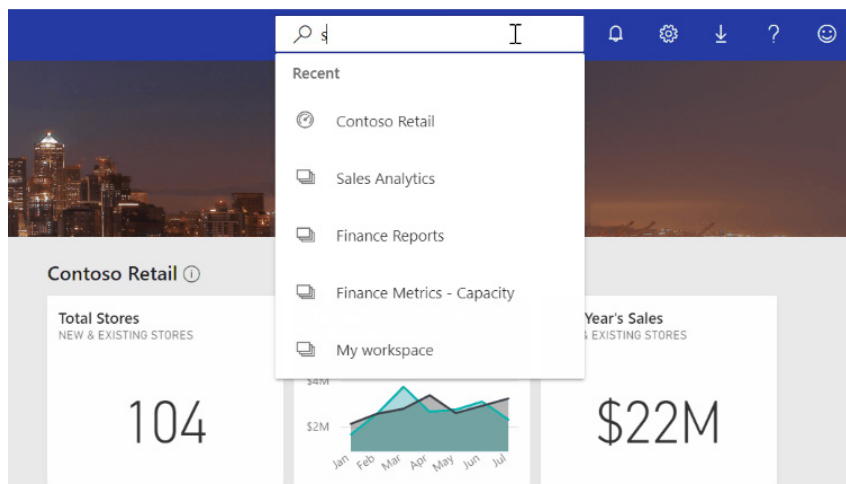
Export report to PDF, PPTX and PNG files using Power BI REST API

In the Power BI Service, users already had the capabilities to export reports to PDF or PPTX. However, this latest update took it a step further by adding an additional layer of capabilities that allows users

to export a Power BI report by using a REST call in the following file formats: PDF, PPTX (PowerPoint) and PNG.

Revamped global search in the Power BI Service

Microsoft has also rolled out a new, revamped search experience that allows users to search for reports, dashboards, apps and workspaces from any page in the Power BI Service.



Performance improvements to Multi Geo Premium capacities

Performance improvements have been made to Premium Capacities configured in remote locations (aka Multi Geo capacities). Now, remote users dependent on the remote capacity for viewing reports no longer need to have their browsers connect to the home region while viewing

and interacting with reports. This will make the reports load faster and the data exploration experience should become entirely more responsive.

Public Preview of read / write XMLA endpoints in Power BI Premium

In late March, the public Preview of read / write capabilities for the XMLA endpoint in Power BI Premium were announced. The read / write capability introduces many additional scenarios for dataset management, advanced semantic modelling, debugging and monitoring.

It is also backwards compatible with Azure Analysis Services and SQL Server Analysis Services tools, making it easier to migrate Analysis Services models to leverage some of the newer capabilities in Power BI.

Updates to On-premises data gateway

In addition to releasing the March update for the On-premises data gateway (version 3000.31.3), enhancements to gateway management operations on Power Platform Admin center were also made, which

includes Gateway Member Status, State and Remove Operation, Gateway Cluster Status, Remove Gateway Clusters and Gateway Cluster Settings.

Updates to Power BI Mobile apps

There have been numerous updates for mobile apps in various platforms. Here's an overview:

INTERACTION SETTINGS

There is now a new section to mobile app settings called Interaction. This section has several items that you can configure to define how the app behaves, including:

- **Docked report footer (phone only):** you may choose whether you want the report footer to stay fixed at the bottom of the report, or to hide and reappear based on your actions in the report, such as scrolling. If you choose to have the footer fixed, it will be hidden only when you view a report in full screen. In all other cases, it will always be visible on the screen, so all actions will be easily available for you
- **Multi select:** when it is turned on, each tap you make on a data point in report is added to previously selected data points, and the combined result is highlighted across all visuals. To deselect a data point, simply tap it again
- **Report refresh (Android phone only):** you may choose how to refresh reports. You can either add a refresh button to the report header or use the pull-to-refresh action (pulling down slightly from top to bottom) in the report page
- **Single tap in reports (phone only):** when single tap is turned on, you tap only once on a visual, button or slicer to start interacting with its data. When single tap is turned off, you have to tap twice: once to select the visual and once to start interacting with its data

PLATFORM SPECIFIC ENHANCEMENTS

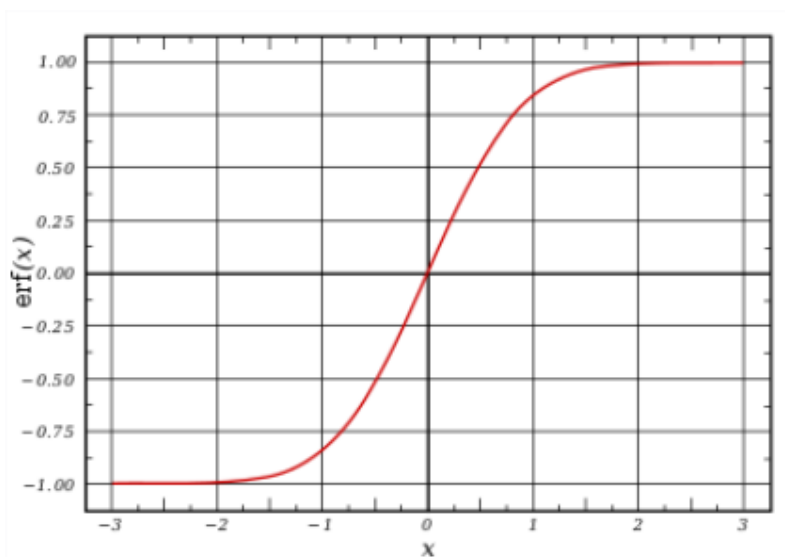
- **Scanning is now available on iPad:** you can now scan barcode to filter reports, or use QR code to quickly open report from iPad
- **Power BI Windows App** now supports PBIX-RS reports
- **Enable Android biometric identification to protect your Power BI data:** use device built-in protection to secure your Power BI data and get the privacy you need. You can now configure your app to require Fingerprint ID or passcode for accessing Power BI, to ensure your confidentiality.

The A to Z of Excel Functions: ERF

In mathematics, the error function (also called the Gauss error function or ERF) is a special, non-elementary function that occurs in probability, statistics and partial differential equations describing diffusion. It is defined as:

$$\begin{aligned}\operatorname{erf}(x) &= \frac{1}{\sqrt{\pi}} \int_{-x}^x e^{-t^2} dt \\ &= \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt.\end{aligned}$$

In statistics, for non-negative values of x , the error function has the following interpretation: for a random variable Y that is normally distributed with mean 0 and variance 1/2, **ERF(x)** describes the probability of Y falling in the range $[-x, x]$.



This function returns the error function integrated between **lower_limit** and **upper_limit**.

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

ERF(lower_limit, [upper_limit])

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

- **lower_limit**: this is required and represents the lower bound for integrating **ERF**
- **upper_limit**: this argument is optional. This represents the upper bound for integrating **ERF**. If omitted, ERF integrates between zero and **lower_limit**.

It should be further noted that:

- if **lower_limit** is nonnumeric, **ERF** returns the **#VALUE!** error value
- if **upper_limit** is nonnumeric, **ERF** returns the **#VALUE!** error value.

Please see the example below:

	A	B	C
1	Formula	Description	Result
2	=ERF(0.5)	Error function integrated between 0 and 0.5	0.5205
3	=ERF(1)	Error function integrated between 0 and 1	0.842701
4			

The A to Z of Excel Functions: ERF.PRECISE

This function is similar to the **ERF** function and was introduced into Excel 2010 for compatibility reasons. The **ERF.PRECISE** function returns the error function similarly.

The **ERF.PRECISE** function employs the following syntax to operate:

ERF.PRECISE(x)

It has the following arguments:

- **x**: this is required and represents the lower bound for integrating **ERF.PRECISE**.

It should be further noted that:

- if **x** is nonnumeric, **ERF.PRECISE** returns the **#VALUE!** error value.

Please see the example below:

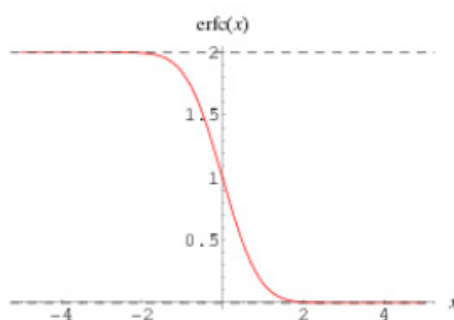
	A	B	C
1	Formula	Description	Result
2	=ERF.PRECISE(0.5)	Error function integrated between 0 and 0.5	0.5205
3	=ERF.PRECISE(1)	Error function integrated between 0 and 1	0.842701
4			

The A to Z of Excel Functions: ERFC

The complementary error function, denoted by either **ERFC(x)** or **erfc(x)**, is defined as

$$\begin{aligned}\operatorname{erfc}(x) &= 1 - \operatorname{erf}(x) \\ &= \frac{2}{\sqrt{\pi}} \int_x^{\infty} e^{-t^2} dt\end{aligned}$$

This function returns the complementary **ERF** function integrated between **x** and infinity.



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

ERFC(x)

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

- **x**: this is required and represents the lower bound for integrating **ERFC**.

It should be further noted that:

- if **x** is nonnumeric, **ERFC** returns the **#VALUE!** error value.

An example:

	A	B	C
1	Formula	Description	Result
2	=ERFC(0.5)	Complementary error function integrated between 0 and 0.5	0.4795
3	=ERFC(1)	Complementary error function integrated between 0 and 1	0.157299
4			

The A to Z of Excel Functions: ERFC.PRECISE

This function is similar to the **ERFC** function and was introduced into Excel 2010 for compatibility reasons.

The **ERFC.PRECISE** function employs the following syntax to operate:

ERFC.PRECISE(x)

The **ERFC.PRECISE** function has the following arguments:

- **x**: this is required and represents the lower bound for integrating **ERFC.PRECISE**.

It should be further noted that:

- if **x** is nonnumeric, **ERFC.PRECISE** returns the **#VALUE!** error value.

Please see our final example below:

	A	B	C
1	Formula	Description	Result
2	=ERFC.PRECISE(0.5)	Complementary error function integrated between 0 and 0.5	0.4795
3	=ERFC.PRECISE(1)	Complementary error function integrated between 0 and 1	0.157299
4			

More Excel Functions next month...

Beat the Boredom Suggested Solution

Earlier, I asked you to consider the following report:

Multiple Criteria with Text Challenge									
Array Data									
Person	Account Type	No of Accounts	Paperwork	Interview	Processing	Postage	Set-up	Overheads	
A	Standard	117	3.4	3	1.2	2.5	1.5	2.1	
A	Deluxe	22	2.1	1.7	3.5	3.8	3.2	1	
B	Deluxe	49	3.4	1.5	1.5	2.9	3	3.3	
B	Standard	68	1.8	4	3.3	2.4	1.8	1.6	
Hourly Rate			\$15	\$24	n/a	\$19	\$29	\$38	
Total Costs									
Person	B								
Account Type	Deluxe								
Total Costs	\$17,370.5								

The aim was to be able to cross-multiply all costs that meet certain criteria. The criteria (Business and Account Type) were provided in cells **C6:C9** and **D6:D9**, with the number of accounts in cells **E6:E9**. The hours required for the various processes per account type and business was in cells **F6:K9** with the hourly rates by process in cells **F11:K11**. Clear as mud?

In summary, if all costs were calculated, you'd need to cross multiply all of the cells, e.g. (117 x 3.4 x \$15) + (117 x 3.0 x \$24) + (117 x... etc.

But I made it harder than that. Firstly, I only wanted to include costs that met certain criteria (Business "B" and account type "Deluxe") and secondly, I decided to make one of the hourly rates "n/a".

In one cell, without adding any helper rows or columns or using different data or VBA, I wanted you to calculate the total costs in one cell – and I even gave one solution, viz.

```
=IFERROR((C6=E16)*(D6=E17)*E6*F6*F11,)+IFERROR((C6=E16)*(D6=E17)*E6*G6*G11,)+IFERROR((C6=E16)*(D6=E17)*E6*H6*H11,)+IFERROR((C6=E16)*(D6=E17)*E6*I6*I11,)+IFERROR((C6=E16)*(D6=E17)*E6*J6*J11,)+IFERROR((C6=E16)*(D6=E17)*E6*K6*K11,)+IFERROR((C7=E16)*(D7=E17)*E7*F7*F11,)+IFERROR((C7=E16)*(D7=E17)*E7*G7*G11,)+IFERROR((C7=E16)*(D7=E17)*E7*H7*H11,)+IFERROR((C7=E16)*(D7=E17)*E7*I7*I11,)+IFERROR((C7=E16)*(D7=E17)*E7*J7*J11,)+IFERROR((C7=E16)*(D7=E17)*E7*K7*K11,)+IFERROR((C8=E16)*(D8=E17)*E8*F8*F11,)+IFERROR((C8=E16)*(D8=E17)*E8*G8*G11,)+IFERROR((C8=E16)*(D8=E17)*E8*H8*H11,)+IFERROR((C8=E16)*(D8=E17)*E8*I8*I11,)+IFERROR((C8=E16)*(D8=E17)*E8*J8*J11,)+IFERROR((C8=E16)*(D8=E17)*E8*K8*K11,)+IFERROR((C9=E16)*(D9=E17)*E9*F9*F11,)+IFERROR((C9=E16)*(D9=E17)*E9*G9*G11,)+IFERROR((C9=E16)*(D9=E17)*E9*H9*H11,)+IFERROR((C9=E16)*(D9=E17)*E9*I9*I11,)+IFERROR((C9=E16)*(D9=E17)*E9*J9*J11,)+IFERROR((C9=E16)*(D9=E17)*E9*K9*K11,)
```

The problem with this solution is (a) it won't scale (e.g. add extra accounts or processes) and (b) even Einstein would struggle to follow it (especially as he's dead).

Here's our suggested solution:

E19 {=SUMPRODUCT(E6:E9*F6:K9*IF(NOT(ISNUMBER(F11:K11)),F11:K11)*(C6:C9=E16)*(D6:D9=E17))}

Person	Account Type	No of Accounts	Paperwork	Interview	Processing	Postage	Set-up	Overheads
A	Standard	117	3.4	3	1.2	2.5	1.5	2.1
A	Deluxe	22	2.1	1.7	3.5	3.8	3.2	1
B	Deluxe	49	3.4	1.5	1.5	2.9	3	3.3
B	Standard	68	1.8	4	3.3	2.4	1.8	1.6

Hourly Rate	\$15	\$24	n/a	\$19	\$29	\$38
-------------	------	------	-----	------	------	------

Total Costs

Person B

Account Type Deluxe

Total Costs \$17,370.5

{=SUMPRODUCT(E6:E9*F6:K9*IF(NOT(ISNUMBER(F11:K11)),F11:K11)*(C6:C9=E16)*(D6:D9=E17))}

Simple!

If that's all you need to know, glad to be of help. For us mere mortals, let me explain below in a bit of prose that Tolstoy would no doubt be proud of. Let me introduce our namesake, the humble **SUMPRODUCT** function. At first glance, **SUMPRODUCT(vector1, vector2, ...)** appears quite ordinary. However, before showing an example, though, look at the syntax carefully:

- A **vector** for Excel purposes is a collection of cells either one column wide or one row deep. For example, **A1:A5** is a column vector, **A1:E1** is a row vector, cell **A1** is a unit vector and the range A1:E5 is not a vector (it is actually an array, but more on that later). The ranges must be contiguous; *and*
- This basis functionality uses the comma delimiter (,) to separate the arguments (vectors). Unlike most Excel functions, it is possible to use other delimiters, but this will be revisited shortly below.

Consider the following sales report:

	A	B	C	D	E	F	G	H
6	Basic Example							
7	Dummy Database							
8								
9								
10								
11						Pricing Point	# of Sales	Total Sales
12						1	14	14
13						2	7	14
14						5	9	45
15						10	5	50
16						25	3	75
17						50	4	200
18						100	8	800
19						250	2	500
20						500	1	500
21						1000	1	1,000
22								
23								3,198
24								

The sales in column H are simply the product of columns F and G, e.g. the formula in cell H12 is simply =F12*G12. Then, to calculate the entire amount cell H23 sums column H. This could all be performed much quicker using the following formula:

=SUMPRODUCT(F12:F21,G12:G21)

i.e. **SUMPRODUCT** does exactly what it says on the tin: it sums the individual products.

	A	B	C	D	E	F	G	H
25	SUMPRODUCT Illustration							
26								
27								
28						Solution		\$3,198
29								

Where **SUMPRODUCT** comes into its own is when dealing with multiple criteria. This is done by considering the properties of TRUE and FALSE in Excel, namely:

- TRUE*number = number (e.g. TRUE*7 = 7); and
- FALSE*number = 0 (e.g. FALSE*7=0).

Consider the following example:

	E	F	G	H
10				
11		Business Unit	Product Type	Sales
12		1	X	\$100
13		1	Y	\$200
14		1	Z	\$300
15		1	Z	\$400
16		2	X	\$500
17		2	X	\$600
18		2	Z	\$700
19		3	Y	\$800
20		3	Z	\$900
21		4	Y	\$1,000
22				

we can test columns **F** and **G** to check whether they equal our required values. **SUMPRODUCT** could be used as follows to sum only sales made by Business Unit 1 for Product Z, viz.

=SUMPRODUCT((F12:F21=1)*(G12:G21="Z")*H12:H21).

For the purposes of this calculation, **(F12:F21=1)** replaces the contents of cells **F12:F21** with either TRUE or FALSE depending on whether the value contained in each cell equals 1 or not. The brackets are required to force Excel to compute this first before cross-multiplying.

Similarly, **(G12:G21="Z")** replaces the contents of cells **G12:G21** with either TRUE or FALSE depending on whether the value "Z" is contained in each cell.

Therefore, the only time cells **H12:H21** will be summed is when the corresponding cell in the arrays **F12:F21** and **G12:G21** are both TRUE, then you will get TRUE*TRUE*number, which equals the said number.

You can get more and more sophisticated:

Notice that **SUMPRODUCT** is not an array formula (i.e. you do not use CTRL+SHIFT+ENTER) but it is what is known as a "pseudo-array" function, so again it can use a lot of memory making the calculation speed of the file slow down.

Note also that this uses the * delimiter rather than the comma, analogous to TRUE*number, etc. If you were to use the comma delimiter instead, the syntax would have to be modified thus:

=SUMPRODUCT(--(F12:F21=1),--(G12:G21="Z"),H12:H21).

Minus minus? The first negation in front of the brackets converts the array of TRUEs and FALSEs to numbers, albeit substituting -1 for TRUE and 0 for FALSE. The second minus sign negates these numbers so that TRUE is effectively 1, rather than -1, whilst FALSE remains equals to zero. This variant often confuses end users which is why I recommend the first version described above.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
18															
19						Costs									
20															
21						Various Data									
22															
23															
24															
25															
26															
27															
28															
29															
30															
31															
32															
33															
34															
35															
36															
37															
38															
39															
40															
41															
42															
43															
44															
45															
46															

Comprehensive Example - Output									
Product costing model for a retail bank									
Test	Type	No. of apps per week	Application Prep (Hours)	Processing time (Hours)	Review Time (Hours)	Materials Prep (Hours)	Back office costs (\$)	Front office costs(\$)	
Transaction account	Standard	80	4.0	1.5	6.0	3.0	\$480	\$250	
Savings account	Budget	45	3.0	0.5	1.0	3.0	\$280	\$200	
Credit card	Budget	50	3.5	0.5	-	1.0	\$1,350	\$100	
Mortgage	Standard	150	5.0	0.8	2.0	3.0	\$400	\$350	
Totals		325	1,380	280	825	875	\$178,500	\$86,500	

Summary Report			
	Total	Budget	Standard
Total cost	\$612,660	\$158,718	\$453,943
Accounts	325	95	230
Avg cost	\$1,885	\$1,671	\$1,974

So far, I have only considered **SUMPRODUCT** with vector ranges. Using the multiplication delimiter (*****), it is possible to use **SUMPRODUCT** with arrays (an array is a range of cells consisting of both more than one row and more than one column).

In the above example, **SUMPRODUCT** has been used in its elementary form in cells **I36:N36**. For example, the formula in cell I36 is:

=SUMPRODUCT(\$H\$32:\$H\$35,I\$32:I\$35)

and this has then been copied across to the rest of the cells.

To calculate the total costs of this retail bank example, this could be calculated as:

=SUMPRODUCT(\$I\$36:\$N\$36,\$I\$21:\$N\$21)

However, the formula in cell **I41** appears more – and unnecessarily – complicated:

=SUMPRODUCT(\$H\$32:\$H\$35*\$I\$32:\$N\$35*\$I\$21:\$N\$21)

The use of the multiplication delimiter is deliberate (the formula will not work if the delimiters were to become commas instead). It should be noted that this last formula is essentially

=SUMPRODUCT(Column_Vector*Array*Row_Vector)

where the number of rows in the **Column_Vector** must equal the number of rows in the **Array**, and also the number of columns in the **Array** must equal the number of columns in the **Row_Vector**.

The reason for this extended version of the formula is in order to divide the costs between Budget and Standard costs in my example. For example, the formula in cell J41 becomes:

=SUMPRODUCT(\$H\$32:\$H\$35*\$I\$32:\$N\$35*\$I\$21:\$N\$21*(G\$32:G\$35=J\$40))

i.e. the formula is now of the form

=SUMPRODUCT(Column_Vector*Array*Row_Vector*Condition)

where **Condition** uses similar logic to the TRUE / FALSE examples detailed earlier. This is a powerful concept – and now you can see how this will answer our question.

Using the above syntax, I can try

Person	Account Type	No of Accounts	Paperwork	Interview	Processing	Postage	Set-up	Overheads
A	Standard	117	3.4	3	1.2	2.5	1.5	2.1
A	Deluxe	22	2.1	1.7	3.5	3.8	3.2	1
B	Deluxe	49	3.4	1.5	1.5	2.9	3	3.3
B	Standard	68	1.8	4	3.3	2.4	1.8	1.6

Hourly Rate						
\$15	\$24	n/a	\$19	\$29	\$38	

Total Costs

Person: B
 Account Type: Deluxe
 Total Costs: #VALUE!

Unfortunately, using the syntax described above, I obtain

=SUMPRODUCT(E6:E9*F6:K9*F11:K11*(C6:C9=E16)*(D6:D9=E17))

But that gives me **#VALUE!** That's because cell **H11** contains the text "n/a". **SUMPRODUCT** doesn't like text (that's the point of our challenge!). **SUMIFS** does. For example:

Colour	Include	Region	Amount
Red	Yes	North	250
Amber	No	South	989
Green	Yes	East	#DIV/0!
Green	No	North	831
Red	No	North	Nil
Red	Yes	North	706
Amber	Yes	West	275
Red	Yes	East	493
Green	Yes	South	986
Green	Yes	West	161
Amber	Yes	East	560
Red	Yes	East	Unsure
Amber	Yes	South	576
Amber	No	North	384
Green	Yes	North	604
Red	No	East	608
Amber	Yes	West	794
Green	Yes	South	210
Amber	Yes	East	359
Red	Yes	South	710
Red	No	North	145
Green	Yes	North	213

Criteria

Colour: Green
 Include: Yes
 Region: South

SUMPRODUCT: #DIV/0!
 =SUMPRODUCT((DataTable[Colour]=\$I\$5)*(DataTable[Include]=\$I\$6)*(DataTable[Region]=\$I\$7)*DataTable[Amount])

SUMIFS: \$1,196.0
 =SUMIFS(DataTable[Amount],DataTable[Colour],\$I\$5,DataTable[Include],\$I\$6,DataTable[Region],\$I\$7)

Do you see **SUMIFS** works even though I have failed to explain either the syntax or the formula? (This art That's because **SUMIFS** only works on vectors and my challenge – deliberately – uses arrays. My solution does employ **SUMPRODUCT** but in an unusual way – and the original example file gave you a massive clue. Did you spot it?

Did you notice "n/a" was highlighted bold italic yellow on red in the original Excel file?

That's because I used conditional formatting. It highlights the cell accordingly if

=NOT(ISNUMBER(F\$11))

is TRUE. **ISNUMBER** determines if a cell or value is a number; **NOT** converts TRUE to FALSE and *vice versa*. Therefore, a cell is highlighted if it does not contain a number.

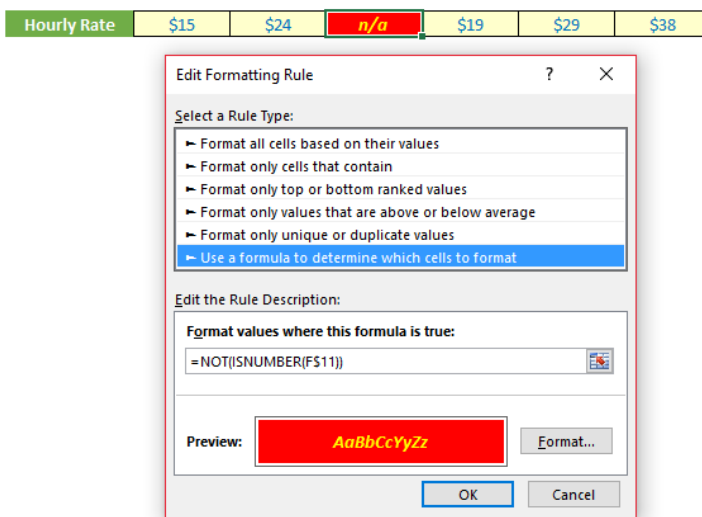
Now consider my solution. It expands on my first attempt for the solution as follows:

{=SUMPRODUCT(E6:E9*F6:K9*IF(NOT(ISNUMBER(F11:K11)),F11:K11)*(C6:C9=E16)*(D6:D9=E17))}

Note that this formula replaces **F11:K11** with **IF(NOT(ISNUMBER(F11:K11)),F11:K11)**. That means text is treated as zero – a nice workaround trick which makes the **SUMPRODUCT** syntax work provided that the formula has been entered as an array (**CTRL + SHIFT + ENTER** to get the 'curly brackets' formally known as braces). (This is a syntax requirement.)

No doubt I will be avalanched with simpler solutions – but hey, it works and it is a solution!

Another one next month!



Upcoming SumProduct Training Courses - COVID-19 update

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

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

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

Location	Course	Date	Duration
Online	Power Pivot, Power Query and Power BI	18 - 20 May 2020	3 Days
Online	Excel Tips and Tricks	25 May 2020	1 Day
Online	Financial Modelling	26 - 27 May 2020	2 Days
Online	Power Pivot, Power Query and Power BI	22 - 24 Jun 2020	3 Days
Online	Excel Tips and Tricks	6 Jul 2020	1 Day
Online	Financial Modelling	8 - 9 Jul 2020	2 Days
Online	Power Pivot, Power Query and Power BI	20 - 22 Jul 2020	3 Days
Online	Excel Tips and Tricks	18 Aug 2020	1 Day
Online	Financial Modelling	19 - 20 Aug 2020	2 Days
Online	Power Pivot, Power Query and Power BI	7 - 9 Sep 2020	3 Days
Online	Power Pivot, Power Query and Power BI	9 - 11 Nov 2020	3 Days
Online	Excel Tips and Tricks	16 Nov 2020	1 Day
Online	Financial Modelling	17 - 18 Nov 2020	2 Days
Online	Power Pivot, Power Query and Power BI	14 - 16 Dec 2020	3 Days

Key Strokes

Each newsletter, we'd like to introduce you to useful keystrokes you may or may not be aware of. This month, we thought we would see whether you have adequate **CTRL** over your keyboard shortcuts. Take a letter, Ms. Jones:

Keystroke	What it does
CTRL + A	Select current region, select all
CTRL + B	Bold (toggle)
CTRL + C	Copy
CTRL + D	Fill down
CTRL + E	Excel 2013 onwards: Flash fill
CTRL + F	Find dialog
CTRL + G	Go To dialog
CTRL + H	Replace dialog
CTRL + I	Italic (toggle)
CTRL + J	Enter an invisible character to denote a line break (carriage return) for Find, Replace and Text to Columns
CTRL + K	Insert hyperlink
CTRL + L	Excel 2007 onwards: create Table; prior versions: create List
CTRL + N	New workbook
CTRL + O	Open workbook
CTRL + P	Print
CTRL + Q	Excel 2016 onwards: Quick Analysis
CTRL + R	Fill right
CTRL + S	Save
CTRL + T	Excel 2007 onwards: create Table
CTRL + U	Underline (toggle)
CTRL + V	Paste
CTRL + W	Close window
CTRL + X	Cut
CTRL + Y	Redo
CTRL + Z	Undo

Special thanks to our reader **René Geppert** who mentioned several new shortcuts that have come out. Thanks René!

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

Our Services

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

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

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

Link to Others

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

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

Any Questions?

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

Training

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

Check out our more popular courses in our training brochure:



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

Sydney Address: SumProduct Pty Ltd, Suite 803, Level 8, 276 Pitt Street, Sydney NSW 2000
New York Address: SumProduct Pty Ltd, 48 Wall Street, New York, NY, USA 10005
London Address: SumProduct Pty Ltd, Office 7, 3537 Ludgate Hill, London, EC4M 7JN, UK
Melbourne Address: SumProduct Pty Ltd, Level 9, 440 Collins Street, Melbourne, VIC 3000
Registered Address: SumProduct Pty Ltd, Level 6, 468 St Kilda Road, Melbourne, VIC 3004

contact@sumproduct.com
www.sumproduct.com
 +61 3 9020 2071