Three quarters of the way through the year and our
newsletter shows no signs of slowing down / shrinking in size. Things continue to run apace (amok?) in the wonderful worlds of Excel and business modelling…

With the Excel Summit South now over, we provide details of equivalent conferences in Bulgaria and Slovenia for those of you in the European bloc. We also provide news on getting back into Microsoft's *Excel Blog* and I even end up on TV elsewhere – although not in a work capacity!

There's also the regulars: Keyboard Shortcuts, Visual Basics, Power Pivot Principles, Power Query Pointers and the plethora of Power BI updates. In fact, one such Power BI update is about a new book out now…

Until next month.

Liam Bastick, Managing Director, SumProduct

## Countdown to 8 out of 10 Cats does SumProduct

They say the cameras put the pounds on you. We estimate there must have been at least 200 cameras on our illustrious director pictured here with Rachel Riley and Susie Dent of Countdown / 8 out of 10 Cats fame…

We thought we would start with something a little different this month… For those clients and / or readers in the UK, SumProduct's very own **Liam Bastick** is to be a contestant on the legendary quiz show *Countdown*, a light-hearted 45 minutes of 30-second numbers and letters challenges. We know some of you might have thought knowing Liam's "legendary" sense of humour he might be on the 8 out of 10 Cats version, but as we painstakingly pointed out to him, you have to be both funny and photogenic to be on that version!

Previously, Liam was on the Australian variant *Letters & Numbers* back in 2011, where he ended up an (undefeated) retiring champ over six episodes. How did he get on in the UK show? Well, it's on Channel 4 this Monday (2 September) if you want to find out. Losers walk away with a dictionary and a goody bag, whereas winners get the same plus a teapot! How delightfully British, as our US editor put it…

Excel Blog

After a six-year hiatus, we are pleased to announce SumProduct has managed a return visit to Microsoft's Excel blog, techcommunity.microsoft.com/t5/Excel-Blog/bg-p/ExcelBlog. We note fellow Excel MVP **Ingeborg Hawighorst** has also posted too, so we'd like to pass on our congratulations to her too:
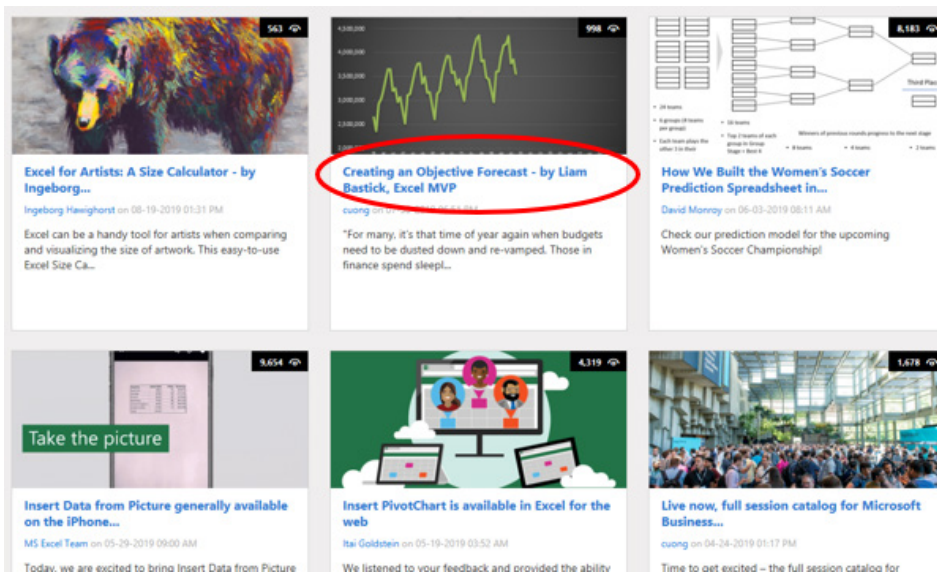
Back in 2013, we posted what became the fourth most visited page of the year – so it's nice to be back! To celebrate, we thought we'd post the new and the old here too, for posterior / posterity!

Creating an Objective Forecast

For many, it's that time of year again when budgets need to be dusted down and re-vamped. Those in Finance spend sleepless nights reviewing minutiae and "normalising" data before talking to their Operational counterparts only to be told it's all wrong and their efforts are wasted. Similarly, the Ops managers see it that Finance descends upon them like stormtroopers in wave formation demanding ridiculous detail to impossible deadlines. It all ends in tears, year after year.

Sound familiar? Well, Excel can help.

The truth is managers and analysts need to work together to improve forecasting. They need to spend more time analysing and less time preparing. Power Query (Get & Transform), Power Pivot and Power BI can all help with that – but that's a story for another day.

What I want to talk about is the need for *objective* forecasting. By this, I mean something that can be constructed simply such that if anyone follows the same process, they will get the *same* figures. This needs to be a mechanical, objective process. This way, analysts may prepare this data in moments without feeling emotionally attached to the outputs. Furthermore, operational managers can review the trend and state where future numbers are wrong and all they need to do is explain the variation, *i.e.* undertake incremental budgeting. There is no need to disagreements or confrontation: both parties may work together as a team.

What we need is a process. As an example, consider the following:
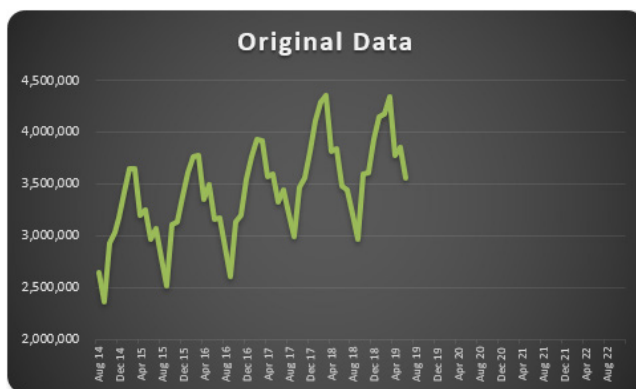
The aim is to develop a technique to identify what would be next in a series, *i.e.* forecast the future. There are various approaches you could use:

- **Naïve method:** this really does live up to its billing – you simply use the last number in the sequence, *e.g.* the continuation of the series 8, 17, 13, 15, 19, 14, … would be 14, 14, 14, 14, … Hmm, great

- **Simple average:** only a slightly better idea: here, you use the average of the historical series, *e.g.* for the continuation of the series 8, 17, 13, 15, 19, 14, … would be 14.3, 14.3, 14.3, 14.3, …

- **Moving average:** now we start to look at smoothing out the trends by taking the average of the last **n** items. For example, if **n** were 3, then the sequence continuation of 8, 17, 13, 15, 19, 14, … would be 16, 16.3, 15.4, 15.9, 15.9, …

- **Weighted moving average:** the criticism of the moving average is that older periods carry as much weighting as more recent periods, which is often not the case. Therefore, a weighted moving average is a moving average where within the sliding window values are given different weights, typically so that more recent points matter more. For example, instead of selecting a window size, it requires a list of weights (which should add up to 1). As an illustration, if we picked four periods and [0.1, 0.2, 0.3, 0.4] as weights, we would be giving 10%, 20%, 30% and 40% to the last 4 points respectively which would add up to 1 (which is what it would need to do to compute the average).

Therefore the continuation of the series 8, 17, 13, 15, 19, 14, … would be 15.6, 15.7, 15.7, 15.5, 15.6, …

All of these approaches are simplistic and have obvious flaws. A slightly more sophisticated method is called **regression analysis**: well, that takes me back! This is a technique where you plot an independent variable on the **x** (horizontal axis) against a dependent variable on the **y** (vertical) axis. "Independent" means a variable you may select (*e.g.* "June", "Product A") and dependent means the result of that choice or selection.

For example, if you plotted your observable data on a chart, it might look something like this:
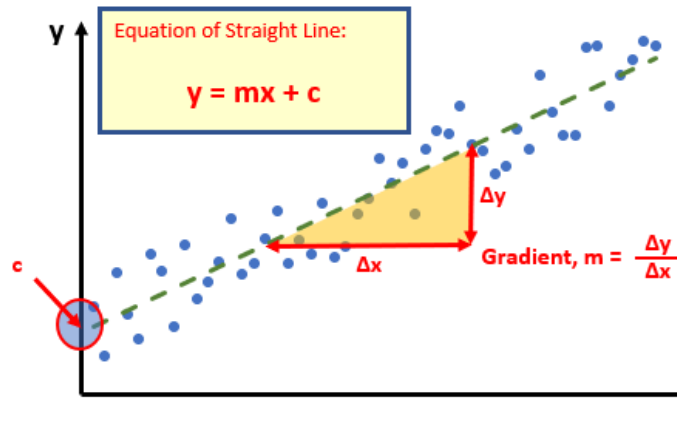
Do you see? You can draw a straight line through the data points. There is a statistical technique where you may actually draw the "best straight line" through the data using an approach such as Ordinary Least Squares, but rather than attempt to explain that, I thought I would try and keep you awake. There are tools and functions that can work it out for you.

Once you have worked it out, you can calculate the gradient (**m**) and where the line cuts the **y** axis (the **y** intercept, **c**). This gives you the equation of a straight line:
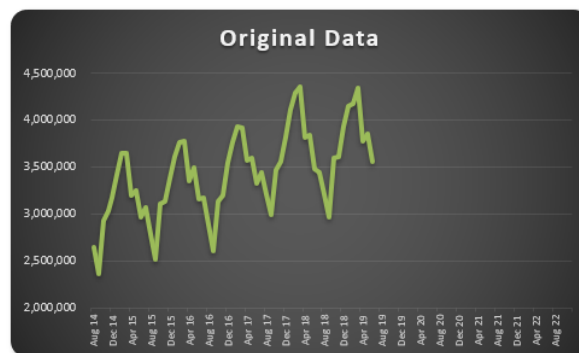
$$y = mx + c$$

Therefore, for any independent value x, the dependent value y may be calculated – and we can use this formula for forecasting.

Of course, this technique looks for a straight line and is known as **linear** regression. You may think you have a more complex relationship (and you may well be right), but consider the following:

- Always split your forecast variables to logical classifications. For example, sales may be difficult to predict as the mix of products may vary period to period, for each product, there may be readily recognizable trends

- If the relationship does not appear to be linear, try plotting **log x** against **log y**. If this has a gradient of two then **y** is correlated with x^2; if the gradient is three, then y is correlated with x^3 *etc*.

With this all borne in mind, let's return to my example:

Now let's be honest, anyone who has historical data looking this perfect should be referred to the auditors, but hey, this is for illustration purposes. I have data from August 2014 until last month. I want to extrapolate it beyond 2020 (I want 2020 foresight!).

There are several functions that can help us here, with one of the simplest being **TREND**. **TREND(known_y's,known_x's,new_x's,[constant])** projects assuming that there is a relationship between two sets of variables **x** (independent variable – here, the dates) and **y** (dependent variable – the sales).

It's preferable to leave **constant** blank in the **TREND** function in order to obtain the best fit. For example:

H72		▼	:	×	✓	fx	=TREND($G$13:$G$71,$F$13:$F$71,$F72)

	E	F	G	H	I	J	S	T
11								
12		Date	Sales	TREND	Combined			
13		30 Aug 14	2,644,539		2,644,539			
14		30 Sep 14	2,359,800		2,359,800			
69		30 Apr 19	3,781,168		3,781,168			
70		31 May 19	3,858,196		3,858,196			
71		30 Jun 19	3,562,680		3,562,680			
72		31 Jul 19		3,921,077	3,921,077			
73		31 Aug 19		3,937,339	3,937,339			
74		30 Sep 19		3,953,077	3,953,077			
75		31 Oct 19		3,969,340	3,969,340			
76		30 Nov 19		3,985,078	3,985,078			
77		31 Dec 19		4,001,340	4,001,340			

Hence, we can extrapolate the data using the **TREND** function *viz*.
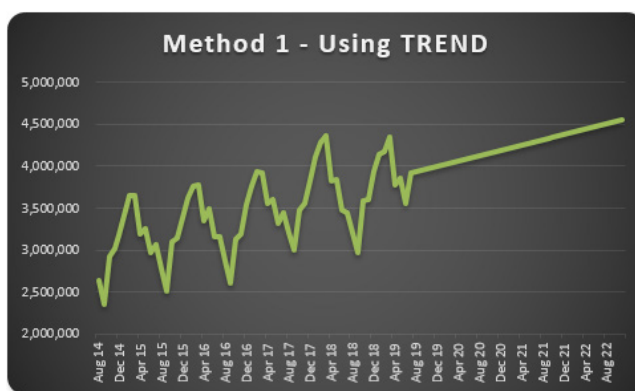
Ladies and gentleman, you may have heard of hockeystick projections; well, let me now present you with the swordfish. You extrapolate linearly, you get a *straight line*. Well duh.

This isn't good enough. We need to identify the cyclicality of the data. It appears to go through a cycle once every 12 months. This might not always be the case, but the concept remains the same even if the periodicity is not annual.

I want to calculate a periodic growth rate objectively. There's various ways I can do this. You might argue with me. That's fine. Feel free to write a brief note and send it to someone who cares. That's the problem here – it's *subjective* until your organisation defines how it is to be measured. Then, everyone follows that process and it becomes *objective*.

In my example, I am going to compare the sum of the sales over the 12 months ending 30 June 2019 with the forecast sales as calculated using **TREND** over the 12 months ending 30 June 2020:

Sum of last actual cycle ending 30 Jun 19	44,617,023
Sum of first forecast cycle ending 30 Jun 20	48,109,459
Growth implicit (CAGR)	7.83%

It is this percentage I will use to grow the forecasts.

I then grow each period's value by its corresponding value in the previous period by this percentage (7.83% here). This gives me a more realistic chart:
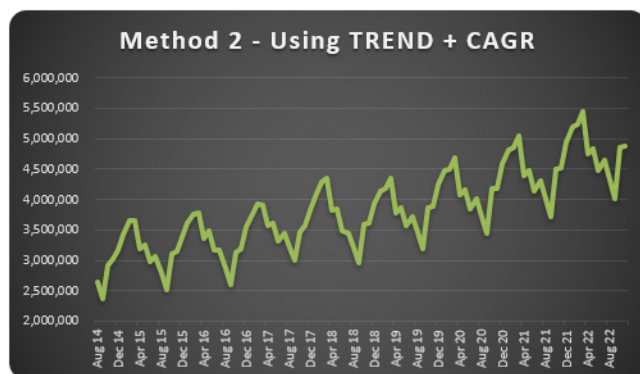
That looks *much* better. With practice, this approach doesn't take that long to prepare. Numbers may be varied from this forecast with the operational manager only having to explain these deviations. It makes life easier all round.

Once the method of assessing inferred growth rates based upon the **TREND** function have been agreed and what normalisations to historical data should be input, the process becomes much more straightforward. Of course, this method should be used for all forecast inputs separately and not just on their aggregation, otherwise confusion occurs due to sales mix changes, new products, cut-off periods, *etc.*

But there is an even faster way – if you happen to have Office 365 or Excel 2016 (or later)…

Exponential Triple Smoothing (ETS) sounds like a dairy process, but it actually uses the weighted mean of past values for forecasting. It's popular in statistics as it adjusts for seasonal variations in data, like in the example above. For those who really need to know, Excel uses a variation of the Holt Winters ETS algorithm, although to be honest, I think you should get out more.

In Excel 2016, ETS has gone "native", *i.e.* it is a standard feature. This includes both a set of new functions such as **FORECAST.ETS** and other supporting functions for additional statistics. Your dataset does not need to be perfect, as the functions will accommodate up to 30% missing data.

But don't worry about using these functions. Simply highlight the actual data and click on the 'Forecast Sheet' button in the 'Forecast' group of the 'Data' tab of the Ribbon **(ALT A + FC)**:
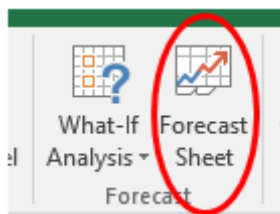
All you need to do is specify the final forecast period at the prompt and that's it. It produces a raw data sheet, together with confidence intervals (to demonstrate potential spread in the forecast error), which looks something like this:
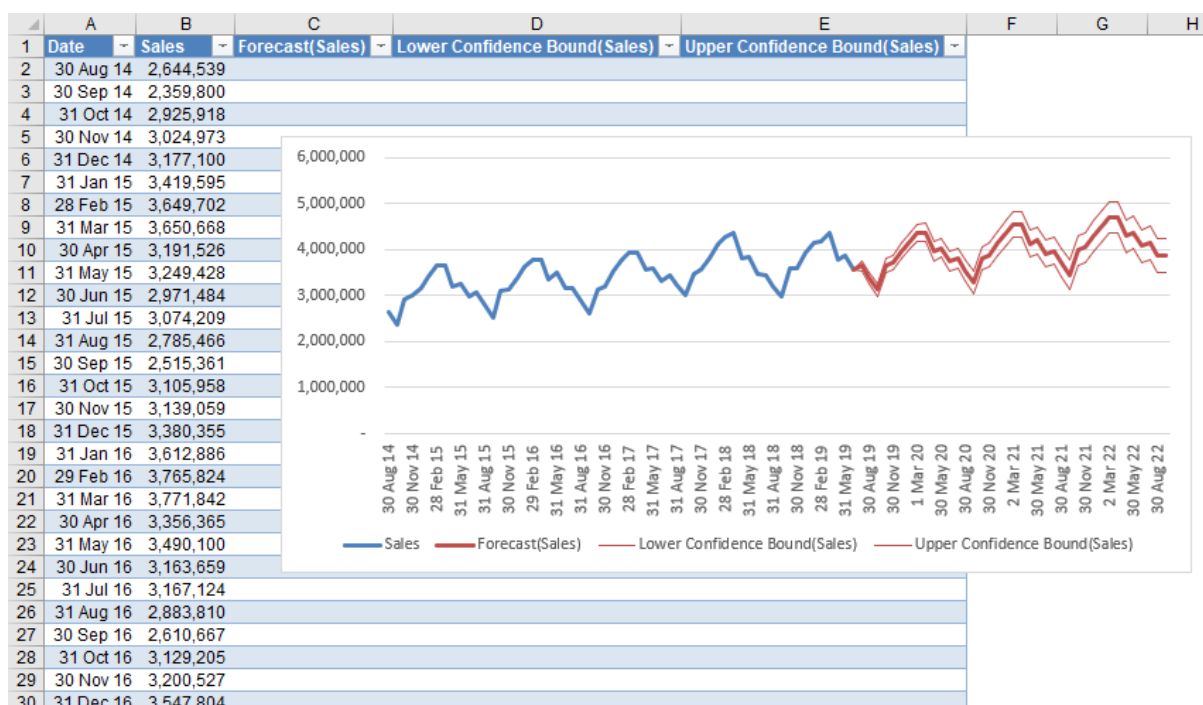
It's objective: you select the data and Excel does all the hard yards for you. If you have explained the concept in layman's terms (as above), then all you need to do is revise the forecast and explain where you wish to change the output from what is generated. Simple!

You may create this longhand. The **FORECAST.ETS** function calculates or predicts a future value based on existing (historical) values by using this Exponential Triple Smoothing (ETS) algorithm. The predicted value is a continuation of the historical values in the specified target date, which

should be a continuation of the timeline. You can use this function to predict future sales, inventory requirements, or consumer trends.

This function requires the timeline to be organised with a constant step between the different points. For example, that could be a monthly timeline with values on the first of every month, a yearly timeline, or a timeline of numerical indices. For this type of timeline, it's very useful to aggregate raw detailed data before you apply the forecast, which produces more accurate forecast results as well.

The **FORECAST.ETS** function employs the following syntax to operate:

FORECAST.ETS(target_date, values, timeline, [seasonality], [data_completion], [aggregation])

The **FORECAST.ETS** function has the following arguments:

- **target_date:** this is required. This is the data point for which you want to predict a value. The **target_date** may be date / time or numeric. If the **target_date** is chronologically before the end of the historical timeline, **FORECAST.ETS** returns the *#NUM!* error
- **values:** this is required. The **values** are the historical values, for which you want to forecast the next points

- **timeline:** this is also required. This is the independent array or range of numeric data. The dates in the timeline must have a consistent step between them and cannot be zero (0). The timeline isn't required to be sorted, as **FORECAST.ETS** will sort it implicitly for calculations. If a constant step cannot be identified in the provided timeline, **FORECAST.ETS** will return the *#NUM!* error. If the **timeline** contains duplicate values, **FORECAST.ETS** will return the *#VALUE!* error. If the ranges of the timeline and values are not of the same size, **FORECAST.ETS** will return the *#N/A* error

- **seasonality:** this argument is optional. This is a numeric value with a default value of 1. This means Excel detects seasonality automatically for the forecast and uses positive, whole numbers for the length of the seasonal pattern. 0 indicates no seasonality, meaning the prediction will be linear. Positive whole numbers will indicate to the algorithm to use patterns of this length as the seasonality. For any other value, **FORECAST.ETS** will return the *#NUM!* error

- The maximum supported **seasonality** is 8,760 (number of hours in a year). Any **seasonality** above that number will result in the *#NUM!* error

- **data_completion:** this argument is also optional. Although the **timeline** requires a constant step between data points, **FORECAST.ETS** supports up to 30% missing data, and will automatically adjust for it. Zero (0) will indicate the algorithm to account for missing points as zeros. The default value of 1 will account for missing points by completing them to be the average of the neighboring points

- **aggregation:** this is the final optional argument. Although the **timeline** requires a constant step between data points, **FORECAST.ETS** will aggregate multiple points which have the same time stamp. The **aggregation** parameter is a numeric value indicating which method will be used to aggregate several values with the same time stamp. The default value of 0 will use **AVERAGE**, while other options are **COUNT**, **COUNTA**, **MAX**, **MEDIAN**, **MIN** and **SUM**.

Function Number	Function
1	AVERAGE
2	COUNT
3	COUNTA
4	MAX
5	MEDIAN
6	MIN
7	SUM

As for me, I'll just press the button and keep in mind the simple idea behind it all. After all, during the forecasting season, I'd still like to go home and see the folks.

Summing Data Across Multiple Criteria on Multiple Sheets

Ever had to sum data based on multiple criteria situated in different Excel worksheets? Here, I provide a quick tour of **INDIRECT** references and Table functionality while combining qualities of the **SUMPRODUCT** function with the **SUMIFS** function to provide a solution to the Mother of all Multiple Criteria problems.

The functionality is best explained by walking through an example. Ivana Car Sales has four divisions, cunningly called North, South, East and West. Each quarter, the four divisions are required to submit sales reports detailing the month of sale, the salesperson, the car color and the price the car was sold for, *viz.*

North Report

Month	Sales Person	Car Colour	Amount ($k)
Jan	Albert	Red	22
Jan	Beattie	White	18
Feb	Beattie	Blue	15
Feb	Albert	Red	15
Feb	Charlie	Red	23
Feb	Delta	Red	20
Feb	Delta	Blue	21
Feb	Albert	Blue	19
Feb	Charlie	White	18
Mar	Charlie	White	12
Mar	Delta	White	14

South Report

Month	Sales Person	Car Colour	Amount ($k)
Jan	Charlie	Blue	15
Jan	Beattie	Red	15
Feb	Beattie	Red	20
Mar	Albert	Red	35
Mar	Charlie	Red	30

East Division Results

Relevant Summary

Month	Sales Person	Car Colour	Amount ($k)
Jan	Charlie	White	27
Jan	Delta	White	10
Jan	Delta	Blue	25
Jan	Albert	Red	30
Jan	Charlie	Blue	10
Jan	Beattie	White	15
Feb	Delta	Blue	21
Feb	Albert	Blue	19
Feb	Charlie	Red	20
Feb	Charlie	White	12
Mar	Delta	White	15
Mar	Beattie	Red	15
Mar	Delta	White	20
Mar	Beattie	White	30
Mar	Delta	Red	13
Mar	Delta	White	18
Mar	Delta	Blue	27

West Division Results

Relevant Summary

Month	Sales Person	Car Colour	Amount ($k)
Mar	Albert	Red	30
Mar	Beattie	White	50
Mar	Beattie	Blue	65

The question is: how can you determine how many red cars Charlie sold in February in total across all four divisions?

The answer would be fairly straightforward if the data were all on one worksheet. For a single criterion, **SUMIF** would cope admirably well, while for several criteria, **SUMPRODUCT** could be used to generate the answer.

Setting up the reports

There are a few considerations for how the reports are set up:

1. It is important that these reports are constructed from a master template - *i.e.* each report has the various report fields in the same corresponding column. This is important for what follows. If this is not the case, you should modify the datasheets accordingly

2. If the four reports are not stored in the same workbook, the data should be imported onto separate worksheets within the workbook where the output report is to be generated.

If we are to refer to multiple datasheets, we need to know the names of these worksheets. For maximum flexibility, I would suggest storing these in a Table (highlight the data, and then, on the **Insert** tab of the Ribbon, select **Table**):

Relevant Divisions ▼
North
South
East
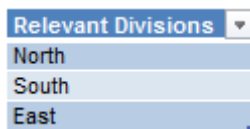
I have named this Table **Division_Table** as this lists the divisions relevant for the analysis. To show why I have used a Table, note that I have not included the West division. If I were to type this into the row beneath East, West would become part of the Table. This is a very useful feature for referencing lists.

It is also important to note that the three divisions named (North, South and East here) <u>must</u> have identical names to the sheet tab names – otherwise, this solution will not work. Ensure that the text is precisely the same as that in the sheet tab.

Multiple Criteria, Multiple Worksheets

We are now ready. In a separate worksheet, I would create the following table:

Multiple Criteria for Relevant Divisions

Results

Month ▼	Sales Person ▼	Car Colour ▼	Total Divisional Results ($k) ▼
Feb	Charlie	Red	43

The formula in cell **I12** here is probably one of the simplest you have ever come across (that is, if you happen to work in the world of Relativistic Quantum Mechanics!):

$$\text{=IFERROR(SUMPRODUCT(SUMIFS(INDIRECT("'"\&Division\_Table[Relevant}$$
$$\text{Divisions]\&"'!I:I"),INDIRECT("'"\&Division\_Table[Relevant}$$
$$\text{Divisions]\&"'!F:F"),\$F12,INDIRECT("'"\&Division\_Table[Relevant}$$
$$\text{Divisions]\&"'!G:G"),\$G12,INDIRECT("'"\&Division\_Table[RelevantDivisions]\&"'!H:H"),\$H12)),)}$$

You know you have created a monster when you nest three complex Excel functions inside a fourth. But have no fear. To work out what is going on, I will explain from the inside out (as this is the order in which Excel will calculate this formula):

- **INDIRECT** – This function produces an array of references such as 'North' column **F**, 'South' column **F**, *etc*. which can be used by the other functions. Note carefully that "'" in the formula is inverted commas, an apostrophe (the syntax required in general for sheet names), then inverted commas.

- **SUMIFS** – This function now applies the multiple criteria to the summation analysis. However, used on its own, this will only report on the first worksheet <u>created</u> (the order of presentation within the workbook is irrelevant). That's where the next function comes in…

- **SUMPRODUCT** – This function is necessary as it is often referred to as a "pseudo array function". What this means in practice here is that it will allow the **SUMIFS** function to be performed across <u>all</u> three worksheets.

- **IFERROR** – This error trap ensures that if a worksheet listed in the **Division_Table** does not exist and/or there is a blank row, the formula will not produce an error - *#REF!* for example.

That's it. Now you'll be able to sum the data across the workbooks and report that Charlie's red car sales totalled $43k in February across the four divisions.

Alternative approaches

Also, there are four other possible solutions to consider: aside from Power Query and Power Pivot, PivotTables using data from multiple worksheets or creating a master data sheet as an interim step, where all data is recorded on one worksheet.

Final thought is: keep it simple, stupid (also known as the KISS rule). Having data on multiple worksheets may complicate the problem and add complexity. Before writing multifaceted formulae like the ones discussed above, always consider simplifying the model structure first.

Bulgaria Excel Days 2019

With 10 lectures by leading world experts in Microsoft Excel, Power BI, Data Analysis and Visualisation, Sofia' Inter Expo Center is the place to be on Thursday 21st November. With six masterclasses and opportunities for business networking, this major event is likely to be another huge success.

Last year saw more than 250 delegates attend – and more is expected this year. For more details and to register, check out the website at www.exceldays.itraining.bg/en/home-en/ .

Excel Olympics Conference & Masterclasses, Slovenia

...And that's not the only event! The following week, Tuesday 26th and Wednesday 27th November sees a similar event at the Austria Trend Hotel in Ljubljana, capital of Slovenia – the only country in the world to have "love" in its name.

Don't let the Crimewatch photo IDs put you off. Four photogenic presenters – plus Leila Gharani – sees another expert group present Excel / Power BI sessions and masterclasses. Should there be sufficient registrations, there may even be a third day!

For more details and to register, check out www.conference.excelolympics.com/?utm_source=EmailSignatureGK&utm_medium=email&utm_campaign=EmailSignatures&utm_content=trustcampaign (one of the most memorable URLs we have seen in a long time!).

Being Sensitive about Excel

Microsoft has undertaken a lot of work in the recent past around collaboration. Sharing files and co-authoring has indeed become easier, but this has led to security considerations. Staff in an organisation need to collaborate both internally and externally. This means that content no longer stays behind a firewall – it roams everywhere, across devices, apps, and services. When it roams, companies need to ensure it is managed in a secure, protected way that meets both business and compliance policies.

With sensitivity labels, you can classify and help protect your sensitive content, while making sure that your people's productivity and ability to collaborate isn't hindered.
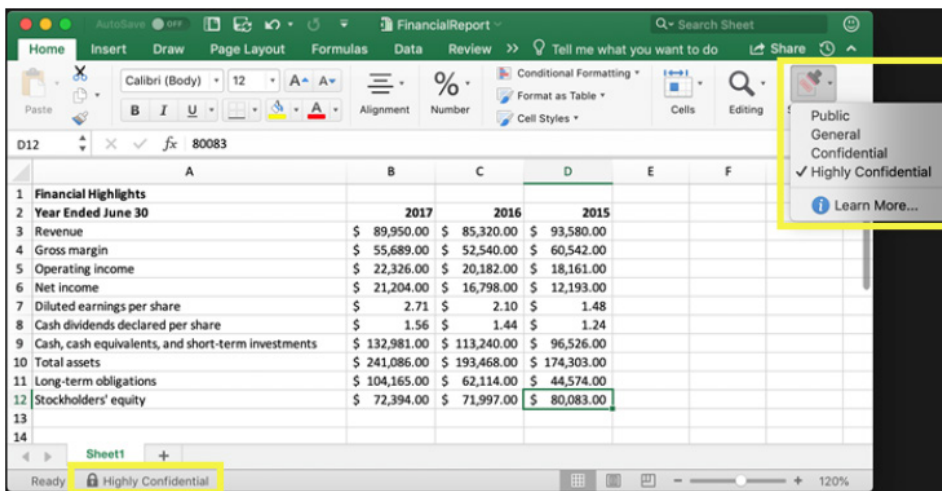
Sensitivity labels may be used to:

- **Enforce protection settings** such as encryption or watermarks on labelled content. For example, users may apply a Confidential label to a document or email, and that label can encrypt the content and apply a Confidential watermark

- **Protect content in Office apps across different platforms and devices.** Sensitivity labels work in Office apps on Windows, Mac, iOS, and Android. Support for Office web apps is coming soon

- **Prevent sensitive content from leaving your organisation on devices running Windows**, by using endpoint protection in Microsoft Intune. After a sensitivity label has been applied to content that resides on a Windows device, endpoint protection can prevent that content from being copied to a third-party app, such as Twitter or Gmail, or being copied to removable storage, such as a USB drive

- **Protect content in third-party apps and services**, by using Microsoft Cloud App Security. With Cloud App Security, you can detect, classify, label, and protect content in third-party apps and services, such as SalesForce, Box, or DropBox, even if the third-party app or service does not read or support sensitivity labels

- **Extend sensitivity labels to third-party apps and services**. With the Microsoft Information Protection software development kit, third-party apps on Windows, Mac, and Linux can read sensitivity labels and apply protection settings. Support for apps on iOS and Android is coming soon

- **Classify content without using any protection settings**. You may also simply assign a classification to content (like a sticker) that persists and roams with the content as it's used and shared. You can use this classification to generate usage reports and see activity data for your sensitive content. Based on this information, you can always choose at a later time to apply protection settings.

With sensitivity labels, you can classify data across your organisation and enforce protection settings based on that classification.

You create sensitivity labels in the Microsoft 365 compliance center, Microsoft 365 security center, or Office 365 Security & Compliance Center. These sensitivity labels can be used by Azure Information Protection, Office apps and Office 365 services.

To be clear, when you assign a sensitivity label to a document or email, it's simply like a tag that is:

- **Customisable:** you can create categories for different levels of sensitive content in your organisation, such as Personal, Public, General, Confidential, and Highly Confidential

- **Clear text:** because the label is in clear text, it's available for third-party apps and services to apply protective actions to label content

- **Persistent:** after a sensitivity label is applied to content, it persists in the metadata of that email or document. This means the label roams with the content, including the protection settings, and becomes the basis for applying and enforcing policies.

In the Office apps, a sensitivity label simply appears as a tag on an email or document. Further, each item of content can have both a single sensitivity label and a single retention label applied to it.
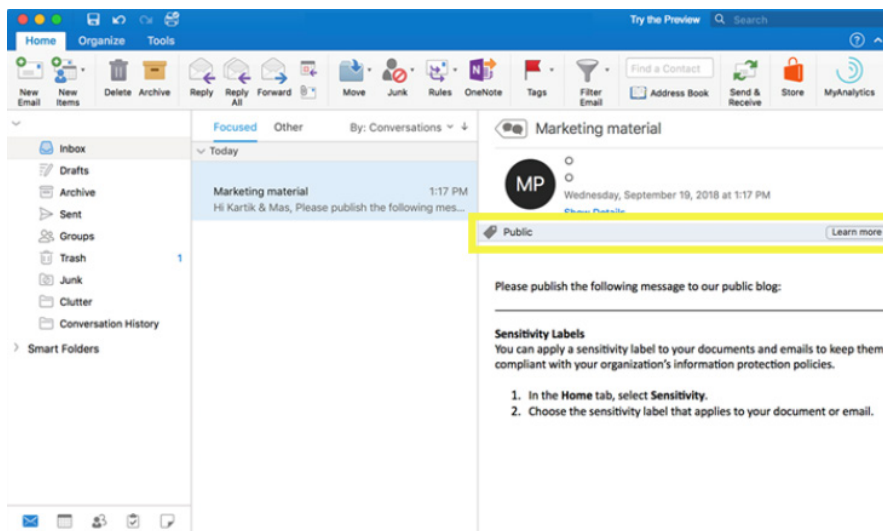
After a sensitivity label is applied to an email or document, the protection settings for that label are enforced on the content. With a sensitivity label, you can:

- **Encrypt email only or both email and documents**. You may choose which users or group have permissions to perform which actions and for how long. For example, you can choose to allow users in a specific domain outside your organisation to have permission to review the content for only seven days after the content is labelled

- **Mark the content** by adding custom watermarks, headers or footers to email or documents that have the label applied. Note that watermarks are applied only to documents, not email, and they're limited to 255 characters. Also, headers and footers are limited to 1,024 characters (except in Excel, where they're limited to 255 characters or fewer, depending on whether the document contains other headers or footers and other factors).

- **Prevent data loss by turning on endpoint protection in Intune**. If sensitive content gets downloaded, you can help prevent the loss of data from Windows devices. For example, you can't copy labelled content into Dropbox, Gmail, or USB drive. Before your sensitivity labels can use Windows Information Protection (WIP), you first need to create an app protection policy in the Azure portal

- **Apply the label automatically to content that contains sensitive information**. You may choose what types of sensitive information that you want labelled, and that label can either be applied automatically, or you can prompt users to apply the label that you recommend.

If you recommend a label, the prompt displays whatever text you choose.
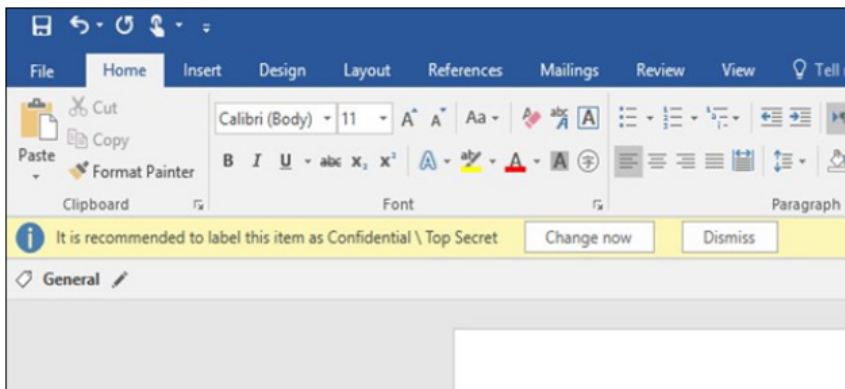
When you create your sensitivity labels, they appear in a list on the Sensitivity tab on the Labels page. In this list, the order of the labels is important because it reflects their priority. You will probably want your most restrictive sensitivity label, such as Highly Confidential, to appear at the bottom of the list, and your least restrictive sensitivity label, such as Public, to appear at the top.

A document or email can have only a single sensitivity label applied to it. If you require your users to provide a justification for changing the label to a lower classification, the order of this list determines what's a lower classification.
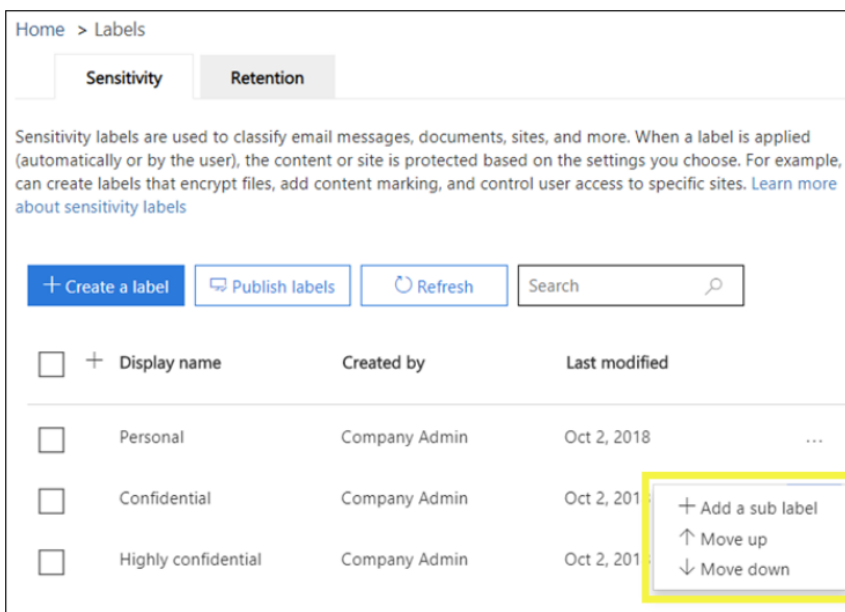
There are things called sublabels too. With sublabels, you can group one or more labels below a parent label that a user sees in an Office app. For example, under Confidential, your organisation might use several different labels for specific types of that classification. In this example, the parent label Confidential is simply a text label with no protection settings, and because it has sublabels, it can't be applied to content. Instead, users must choose Confidential to view the sublabels, and then they can choose a sublabel to apply to content.

Sublabels are simply a way to present labels to users in logical groups. Sublabels don't inherit any settings from their parent label. Sublabels can be applied to content; parent labels cannot.
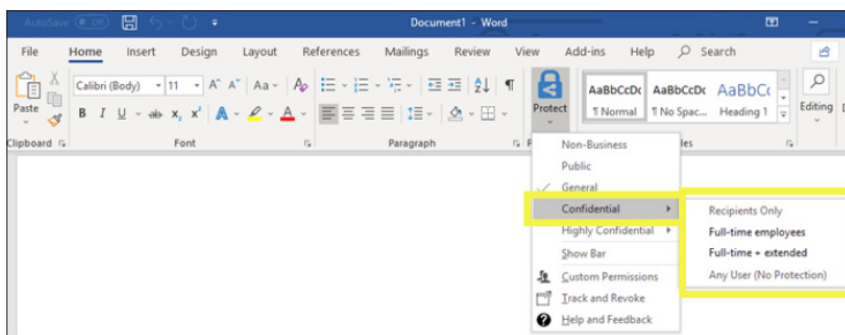
If you delete a sensitivity label, note that the label is not removed from content, and any protection settings continue to be enforced on the content. If you edit a sensitivity label, the version of the label that was applied to content is what's enforced on that content.

After you create your sensitivity labels, you need to publish them, to make them available to people in your organisation, who can then apply the labels to content. Unlike retention labels, which are published to locations, such as all Exchange mailboxes, sensitivity labels are published to users or groups. Sensitivity labels then appear in Office apps for those users and groups.

With a label policy, you can:

- **Choose which users and groups see the labels**. Labels MAY be published to any email-enabled security group, distribution group, Office 365 group or dynamic distribution group
- **Apply a default label** to all new documents and email created by the users and groups included in the label policy. This default label can set a base level of protection settings that you want applied to all your content
- **Require a justification for changing a label**. If content is marked Confidential and a user wants to remove that label or replace it with a lower classification, such as a label named Public, you can require that the user provide a justification when performing this action.

These justifications will be available for the administrator to review
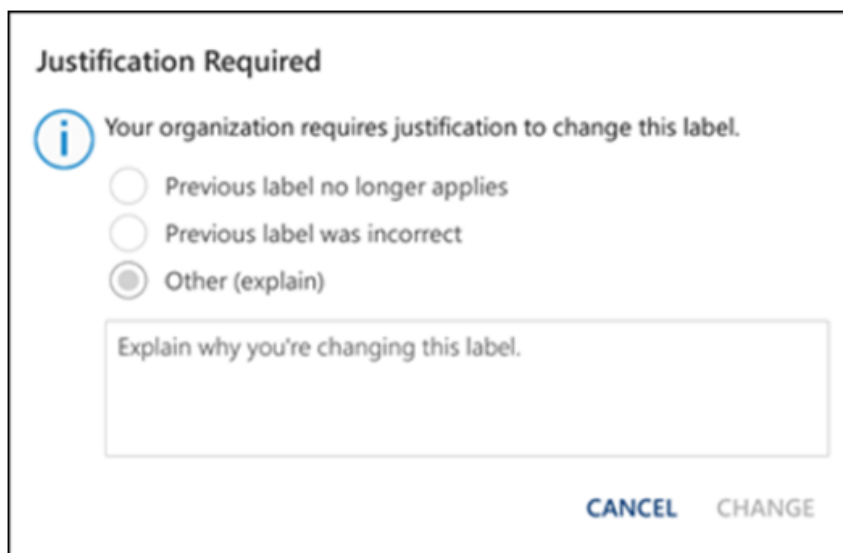
- **Require users to apply a label to their email and documents**. If you want all of a user's content to be labelled, you can require that a label must be applied to all of their saved documents and sent emails. The label can be assigned manually by the user, automatically as a result of a condition or be assigned by default (the default label option described above). Here's the prompt shown in Outlook when a user is required to assign a label:
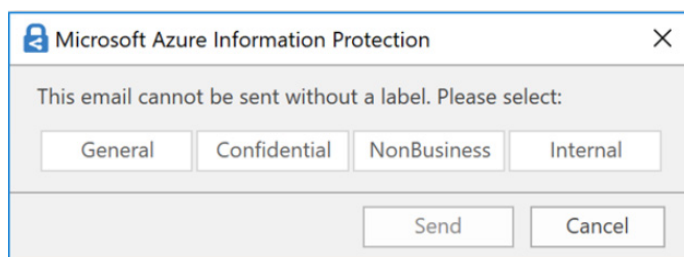
- **Provide help link to a custom help page**. If your users aren't sure what your sensitivity labels mean or how they should be used, you can provide a Learn More URL that appears at the bottom of the Sensitivity label menu in the Office apps.
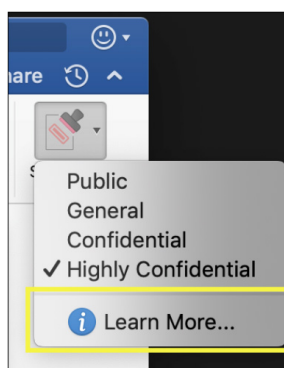
You make your sensitivity labels available to users by publishing them in a sensitivity label policy, which appears in a list on the Sensitivity policies tab on the Label policies page. Just like sensitivity labels, the order of the sensitivity label policies is important because it reflects their priority. The label policy with lowest priority is shown at the top, and label policy with the highest priority is shown at the bottom.

A label policy consists of:

- A set of labels
- The scope of the label policy, meaning the users and groups included in the policy
- The settings of the label policy described above (default label, justification, mandatory label and help link.

You can include a user in multiple label policies, and the user will see all of the sensitivity labels from those policies. However, a user will see the policy settings from only the label policy with the highest priority.

Getting started with sensitivity labels is a quick process:

- **Define the labels**. First, you want to establish your taxonomy for defining different levels of sensitive content. You should use common names or terms that make sense to your users. For example, you can start with labels such as Personal, Public, General, Confidential and Highly Confidential. You can use sublabels to group similar labels by category. Also, when you create a label, a ToolTip is required, which appears in the Office apps when a user hovers over a label option on the Ribbon

- **Define what each label can do**. Second, configure the protection settings you want associated with each label. For example, lower sensitivity content (a "General" label) might simply have a header or footer applied to it, while higher sensitivity content (a "Confidential" label) may have a watermark, encryption and Windows Information Protection (WIP) applied to it, to help ensure that only privileged users can access it

- **Define who gets the labels**. Third, after you define your organisation's labels, you publish them in a label policy that controls which users and groups see those labels. A single label is reusable – you define it once, and then you can include it in several label policies assigned to different users. However, in order for a label to be assigned to content, you must first publish that label so that it's available in Office apps and other services. When just starting out, you can pilot your sensitivity labels by assigning them to just a few people.

Here's the basic flow of what the admin, user, and Office app do to make sensitivity labels work:
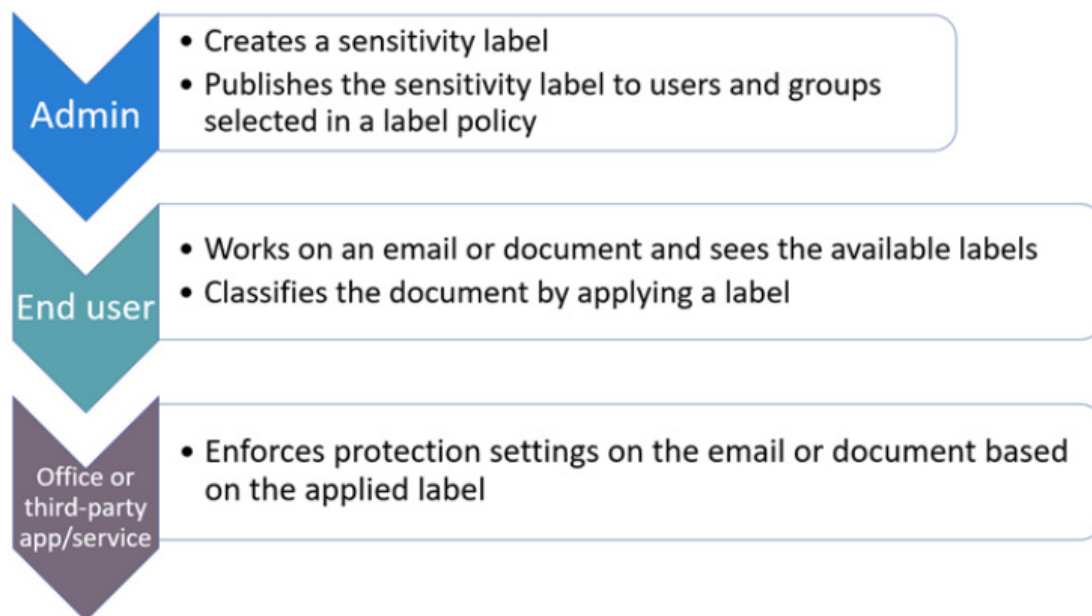
When you create a sensitivity label, you have the option to tell Windows that files with this label are sensitive and need to be protected against data leakage when stored on Windows devices. This option can help ensure that content with this label can be shared or copied only to sanctioned locations, even when it's stored on an endpoint. In essence, turning on this option for a sensitivity label tells Windows that this is extra critical data that warrants additional usage constraints.

When you turn on this option, Windows can read, understand, and act on sensitivity labels in documents and automatically apply WIP on content, no matter how it reaches a managed Windows device. This helps protect labelled files from accidental leakage, with or without applying encryption.

For example, Windows can understand that a Word document residing on a user's machine has a Confidential label applied to it, and WIP can apply an app protection policy to prevent the copying or sharing of the data to any non-work location from that device (such as a personal OneDrive, personal email accounts, social media, or USB drives).

If a user attempts to upload labelled content to a personal Gmail account, they see this message.
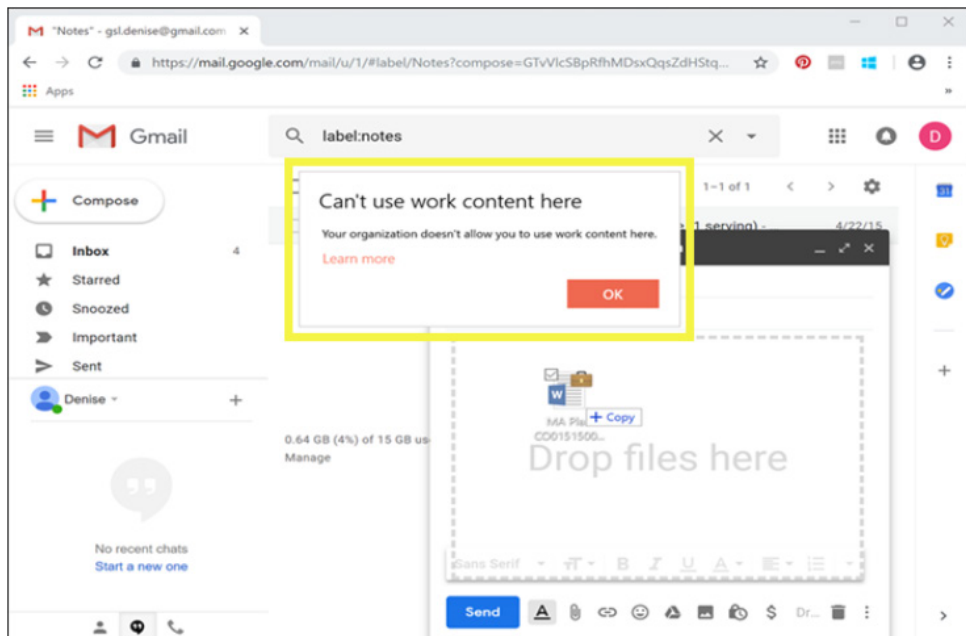
And if a user attempts to save labelled content to a USB drive, they see this message:
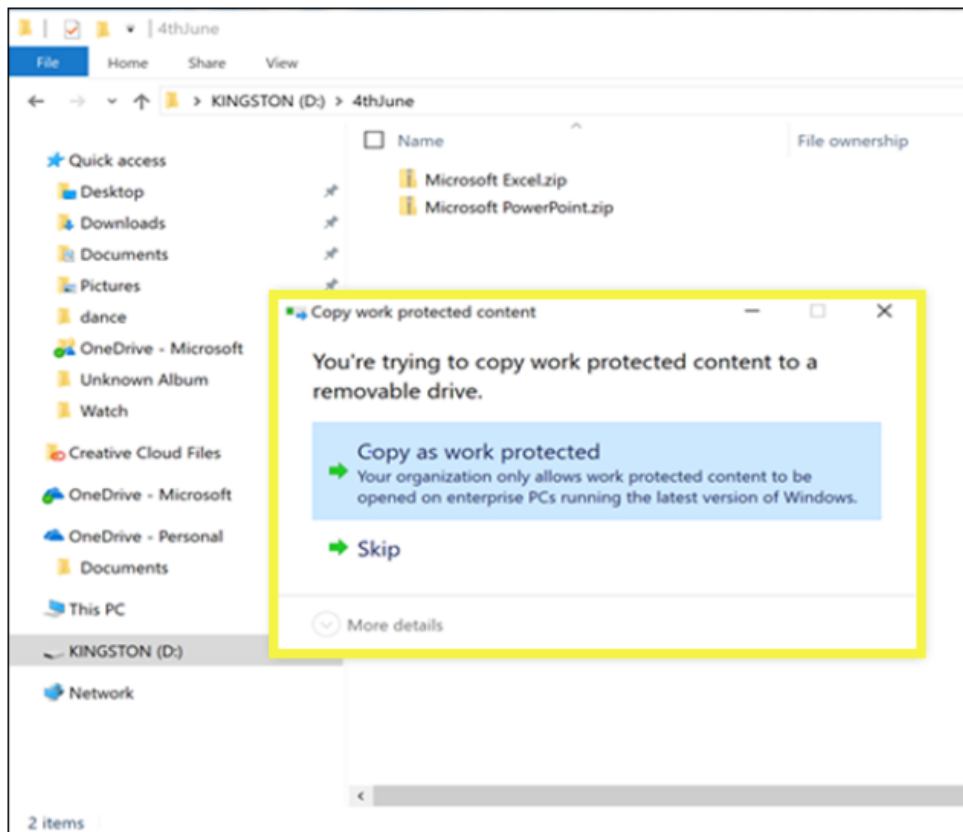
Before your sensitivity labels can use WIP, you first need to do the prerequisites described here:

- Make sure you're running Windows 10, version 1809 or later
- Set up Microsoft Defender Advanced Threat Protection (Microsoft Defender ATP), which scans content for a label and applies the corresponding WIP protection. ATP performs some actions independently from WIP, such as reporting anomalies
- Create a Windows Information Protection (WIP) policy that applies to endpoint devices. You can do this in either of these locations:
  - o Create a Windows Information Protection (WIP) policy with MDM using the Azure portal for Microsoft Intune
  - o Create and deploy a Windows Information Protection (WIP) policy using System Center Configuration Manager.

Visual Basics

We thought we'd run an elementary series going through the rudiments of Visual Basic for Applications (VBA) as a springboard for newer users. This month, we continue our look at control structures, and selection control structures in particular.

In a programming, a control structure determines the order in which statements are executed. Control structures can be grouped into three main categories:

1. **Sequential:** Sequential execution is where each statement in the source code will be executed one by one in a sequential order. This is the default mode of execution
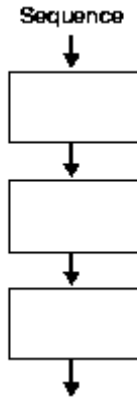
2. **Selection:** The selection control structure is used for making decisions and branching statements
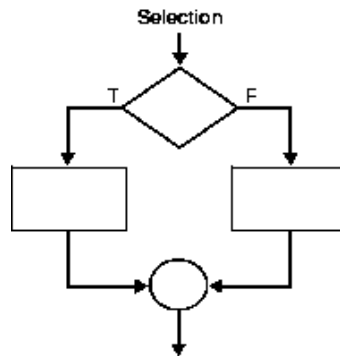
3. **Iteration:** The selection control structure is used for making decisions and branching statements
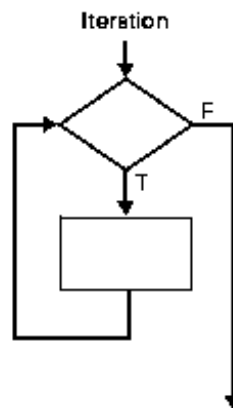
Here are several examples.

SELECT CASE

SELECT CASE is the VBA equivalent to a switch control structure. A switch statement allows a variable to be tested for equality against a list of values. Each value is called a case, and the variable being "switched" on is checked for each case:
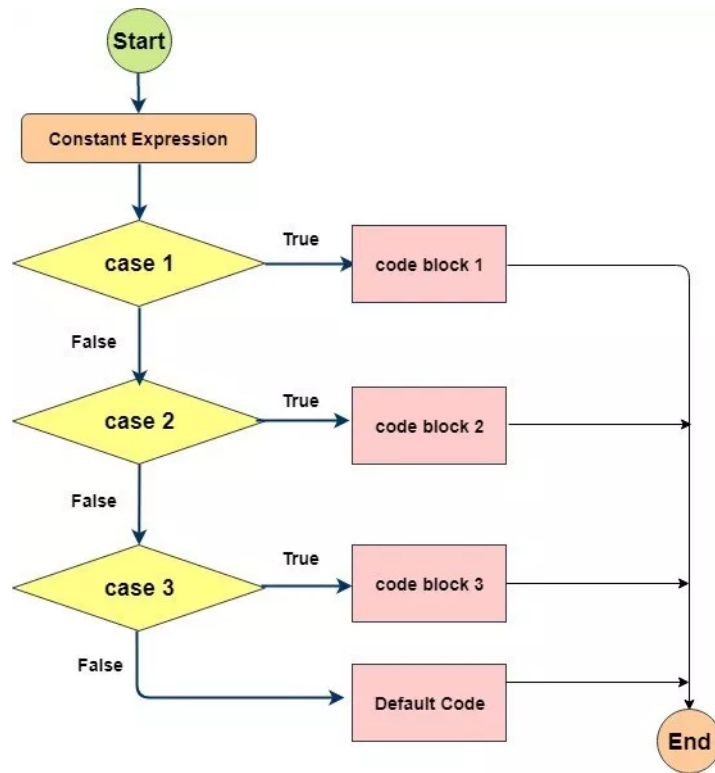
The **SWITCH** statement is used as follows:

```
Select [ Case ] testexpression
    [ Case expressionlist
        [ statements ] ]
    [ Case Else
        [ elsestatements ] ]
End Select
```

Let's look at a coding example:

```
Sub SWITCHStatement()
    Dim myNumber As Integer
    myNumber = 1
    Select Case myNumber
        Case 0
            Debug.Print "Zero the hero!"
        Case 1
            Debug.Print "Number one!"
        Case 2
            Debug.Print "Number two!"
        Case 3
            Debug.Print "Number three!"
        Case Else
            Debug.Print "Not a podium finish"
    End Select
End Sub
```

IS

SELECT CASE allows the use of the **IS** keyword to compare values. If the variable can use comparison operators, then **IS** is used before the operator. For example:

```vba
Sub SwitchISStatement()

    Dim myNumber As Integer

    myNumber = 1

    Select Case myNumber

        Case Is < 0

            Debug.Print "Negative"

        Case Is < 10

            Debug.Print "Single Digit Positive Integer"

        Case Is < 100

            Debug.Print "Double Digit Positive Integer"

        Case Else

            Debug.Print "Large number!"

    End Select

End Sub
```

Note: upon typing an operator (>, < or =) after the keyword **CASE**, the VBA editor will automatically correct the statement and place the **IS** keyword after **CASE**.

TO

If the variable assessed can lie within a range, the **TO** keyword is called upon to denote the range. For example:

```vba
Sub SwitchTOStatement()

    Dim myNumber As Integer

    myNumber = 1

    Select Case myNumber

        Case Is < 0

            Debug.Print "Negative"

        Case 0 To 9

            Debug.Print "Single Digit Positive Integer"

        Case 10 To 99

            Debug.Print "Double Digit Positive Integer"

        Case Else

            Debug.Print "Large number!"

    End Select

End Sub
```

Combination

Multiple expressions can be used in a single **CASE** statement for optimum efficiency. Another example:

```vba
Sub SwitchCombinationStatement()

    Dim myNumber As Integer

    myNumber = 1

    Select Case myNumber

        Case Is < 0

            Debug.Print "Negative"
```

```
        Case 1, 2, 3, 4, 5 To 9

            Debug.Print "Single Digit Positive Integer"

        Case 10 To 50, 51, 52, 53, 54, 55 To 99

            Debug.Print "Double Digit Positive Integer"

        Case Else

            Debug.Print "Large number!"

    End Select

End Sub
```

More next month.

Power Pivot Principles

We continue our series on the Excel COM add-in, Power Pivot. This month, we introduce the syntax differences between a table, a field and a measure in DAX.

Before we go into **D**ata **A**nalytics E**X**pressions (DAX), we first have to learn how to refer to a Table, a Field and a Measure in DAX.

Imagine that we have a table such as the following:

In DAX:

- If the **table** above is called 'Sales' we would refer to it as 'Sales' in DAX
- If we want to refer to any of the **fields** in this table, 'Sales ID' or 'Sales Amount', we would have to write it as 'Sales[Sales ID]', or 'Sales[Sales Amount]'
- If we create a **measure** (*i.e.* a formula which manipulates the aggregated summaries of one or more fields or other calculations) to calculate tax and call it 'Sales Tax', we will refer to it as [SalesTax] with square brackets around it. The reference to a table is not required (and would be confusing) as calculations based upon fields are portable (*i.e.* they may be transferred from one table to another).

This may be summarised as follows:

Table	Sales
Field	Sales[Sales Amount]
Measure	[Sales Tax]

We'll move this on next month, but this is an action-packed newsletter...

Power Query Pointers

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

Last month we looked at how parameters were a possible source when defining conditional columns. This time, let's take a closer look at parameter creation by using a parameter in a filter.

Parameters allow us to enter limits into a calculation or query. If we choose to only show data where the value in my date column matches a parameter, then only data corresponding to the parameter value will be shown. We can then change the parameter to alter the filter.

Here, let's create a parameter using the menu options in the Power Query editor. We begin by entering a new parameter and investigating my options. We can do this from the query editor by choosing the 'Home' tab and picking 'New Parameter' from the 'Manage Parameters' dropdown in the 'Parameters' section.
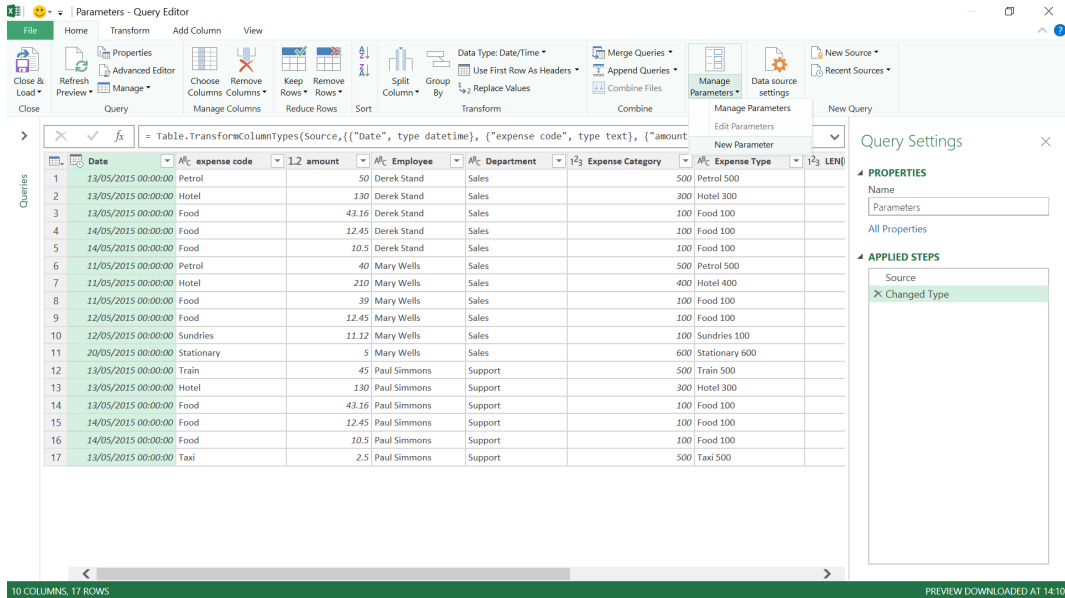
Sticking with my date theme, let's choose to create a parameter that we can use as a date filter. We specify 'Date/Time' to match the data in my *Date* column. The other data types available are shown in the dropdown in the screen below:
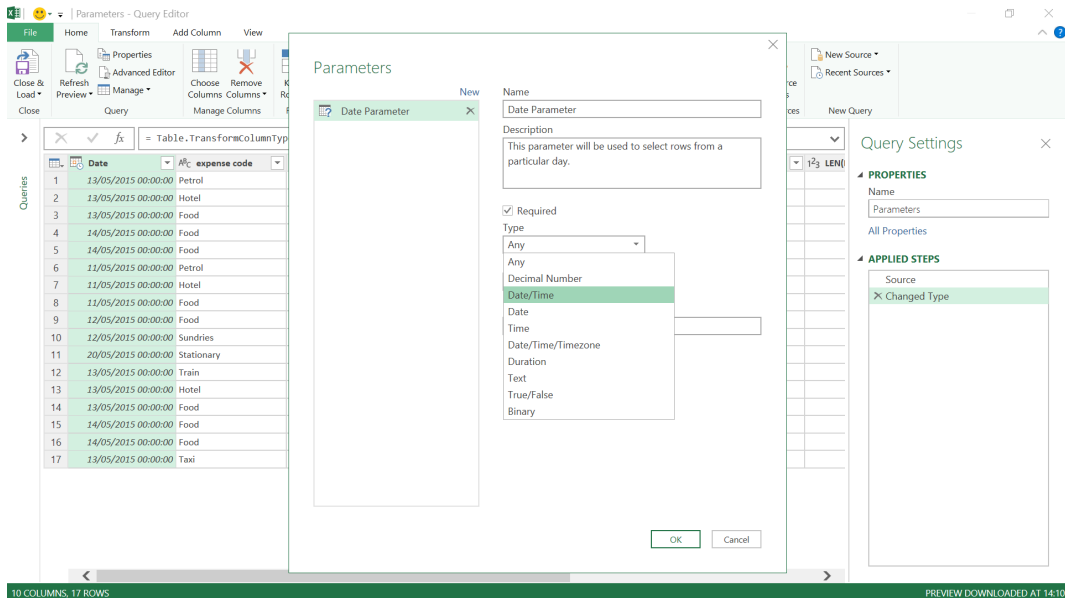
The 'Required' box determines if this parameter must always be populated: if it is ticked then the 'Current Value' must be specified. The next box, 'Suggested Values', defines what can be entered into my parameter. The first option, 'Any value', is obvious enough; the other options need a little more explanation.
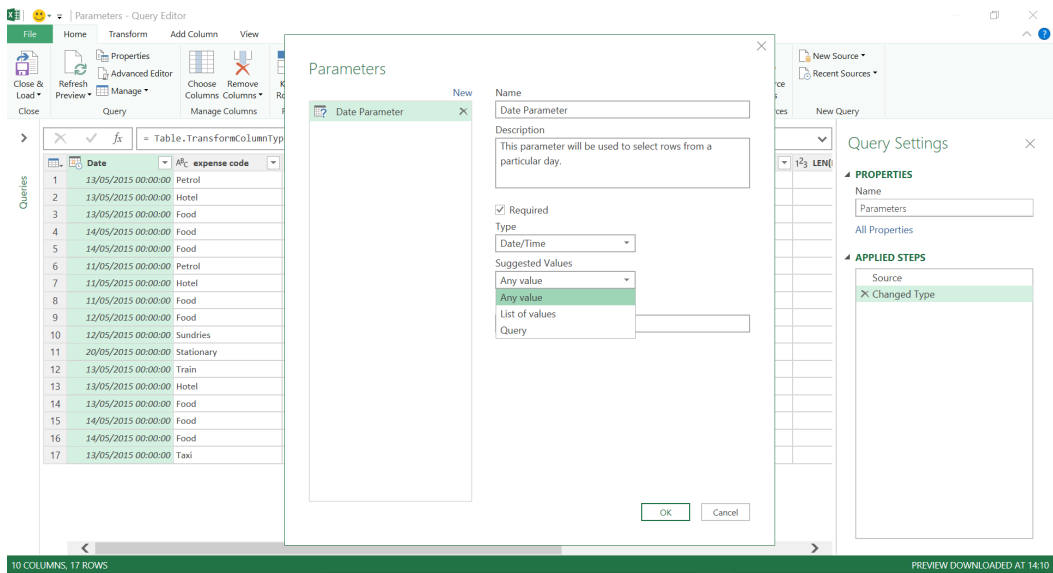
If we choose to only allow a parameter selected from a list ('List of values'), we can limit the options that can be entered into our parameter by creating a list as shown below:
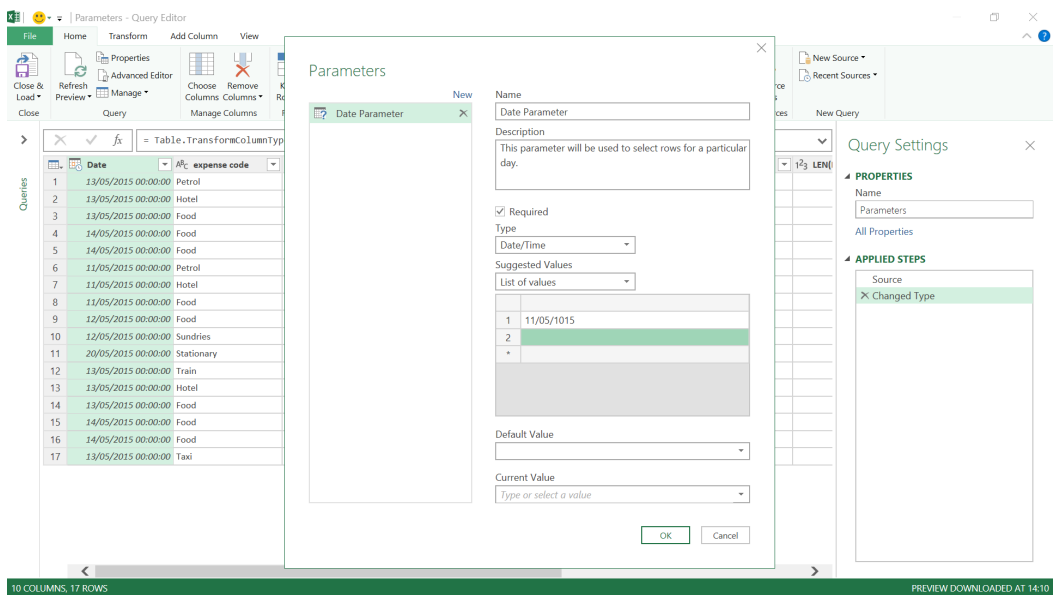
This could help to prevent us from entering a value that wouldn't link to any rows. We can also use a query to determine our subset of dates, but it must have been created prior to creating the parameter.

In order to have something available in this 'Query' option, we need to enter a list query, i.e. a query that will return a list of values, in this case dates. We could create a list using **M** code, but since we want to create a

list of the dates that have data from my query associated with them, we will create the list from my Excel data here.

We create a new query from the Excel data and delete everything except the date column. On the 'Transform' tab, there is an option in the 'Any Column' section to convert a column to a list, called 'Convert to List', as shown below:
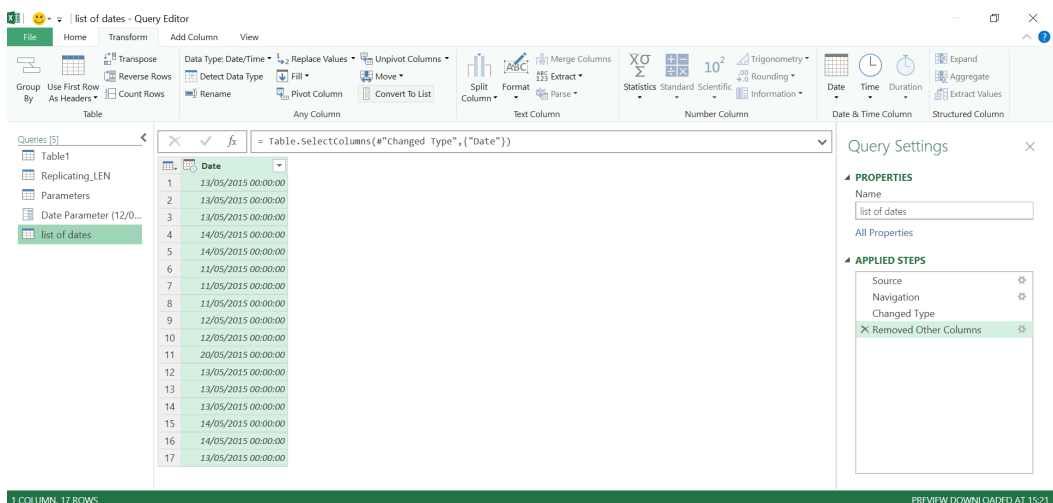
When you click on 'Convert to List', the view changes and the icon for our query changes to a list *viz*.
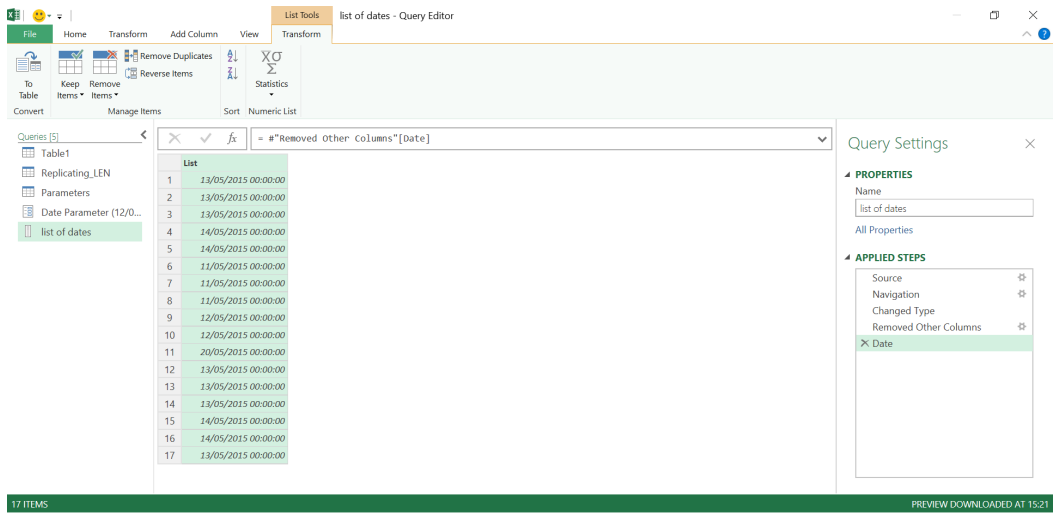
The list is not yet finished: we have dates appearing multiple times in random order. We can tidy up the list by applying a sort and removing duplicates as shown below:
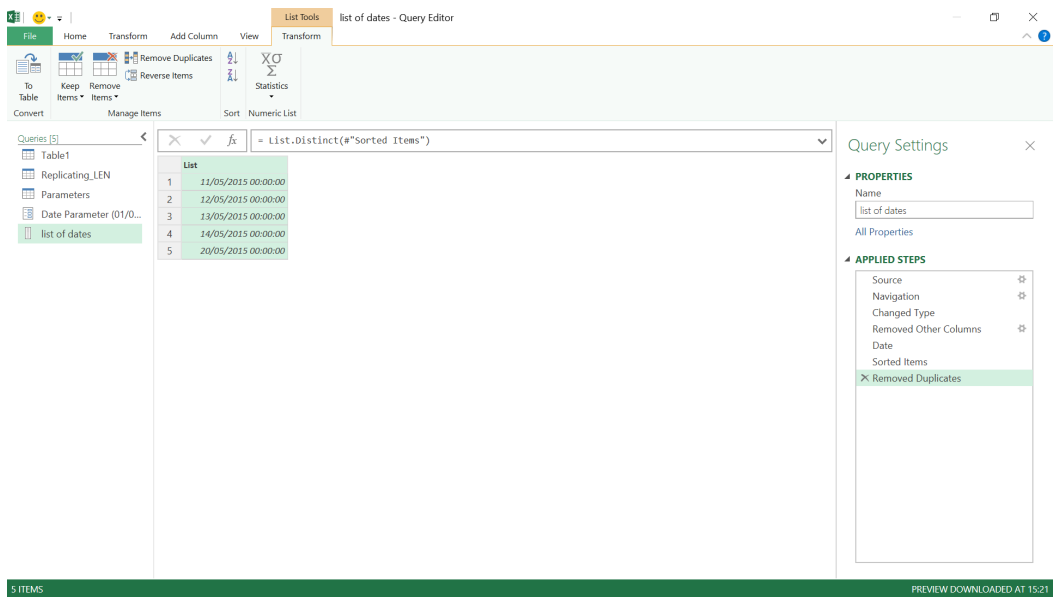
Our list query is now available for us to select when creating or editing the parameter.
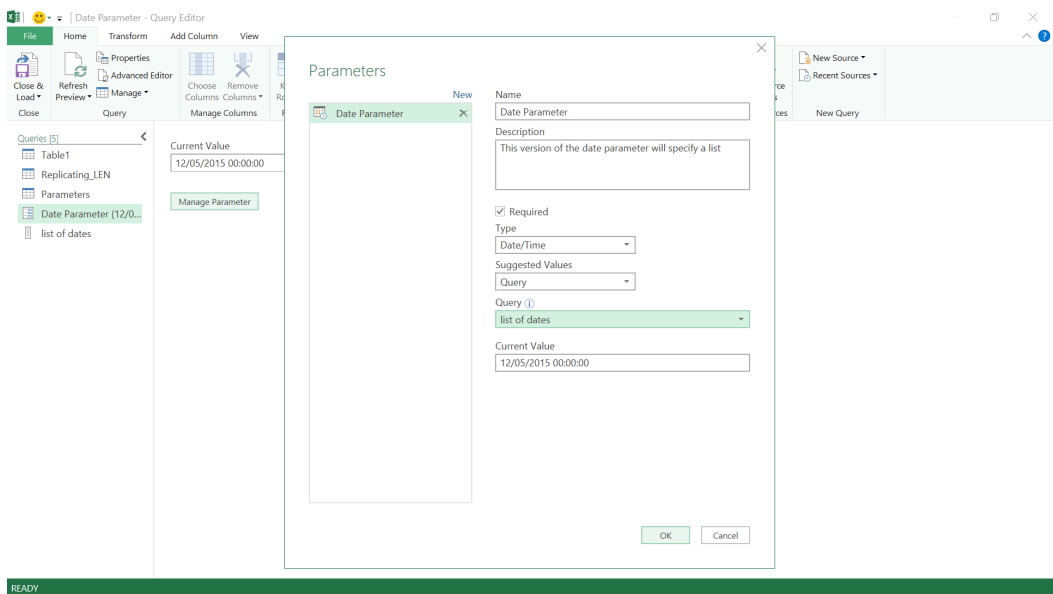
Therefore, we can either create a parameter that accepts any value or have a parameter that will only allow values on a list, whether that is a manually created list in my 'Parameters' screen or via a list query. To see how the parameter can be used in a filter, we begin with the first 'Suggested Values' option, the 'Any value' option:
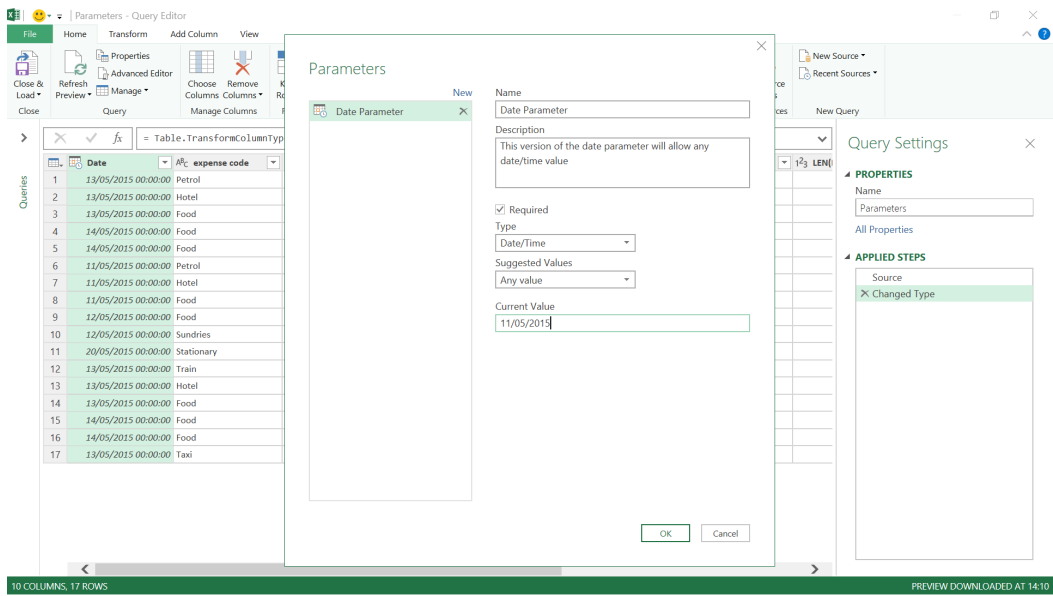
In order to use the parameter to filter our data, we need to go to the filter icon next to the **Date** column, where we find a 'Date/Time Filters' dropdown and pick the 'Custom Filter' option at the bottom.
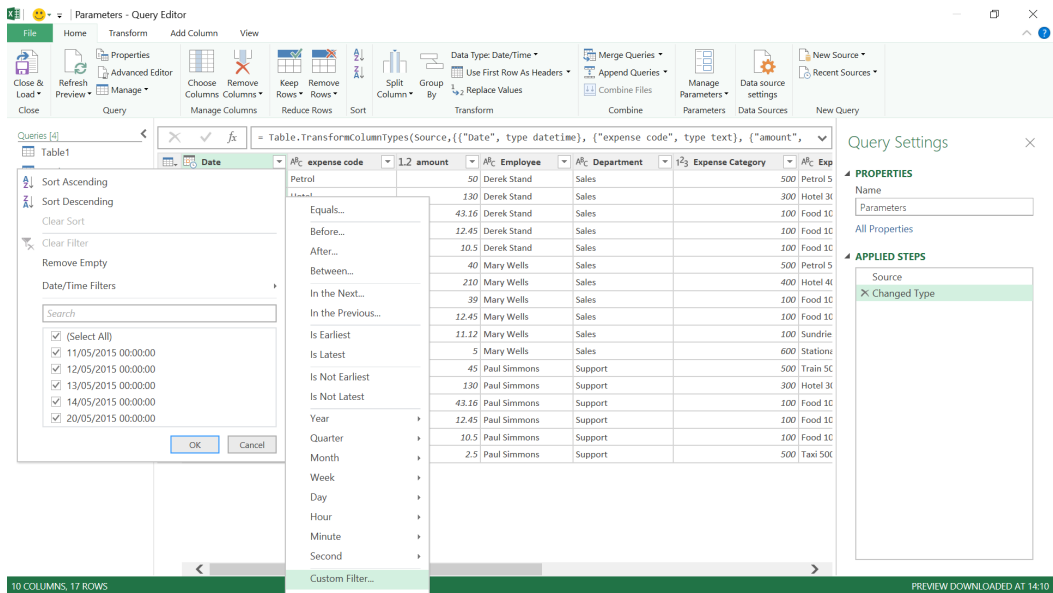
This allows us to filter based on our new parameter by clicking on the datatype icon between the values.
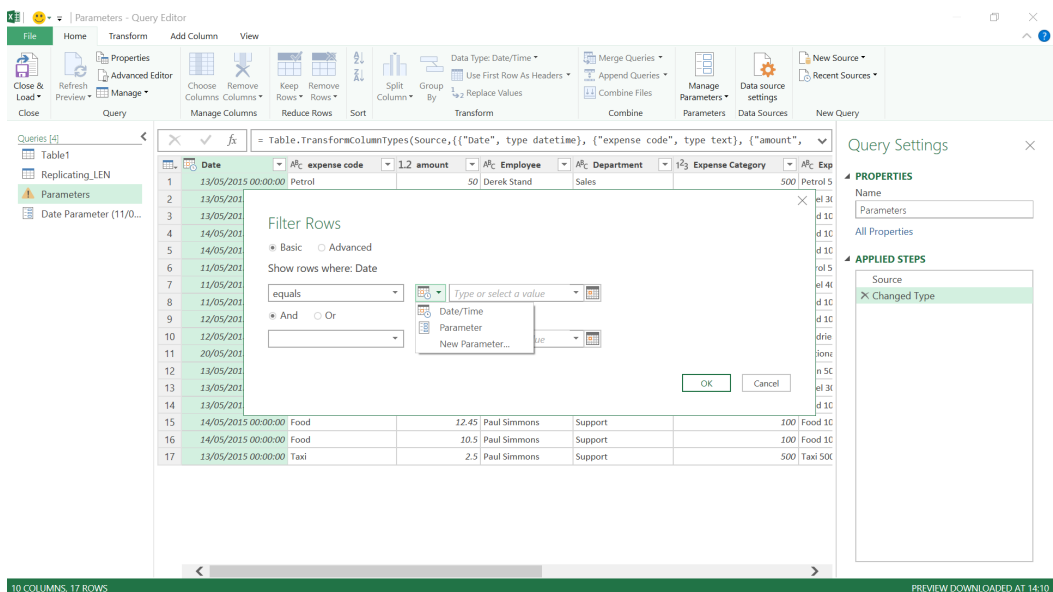
It is useful that we can also enter new parameters from here. However, this is not an option to enter the *first* parameter in the workbook, as the datatype icon is not available if there are no parameters. Let's choose our parameter and click OK.
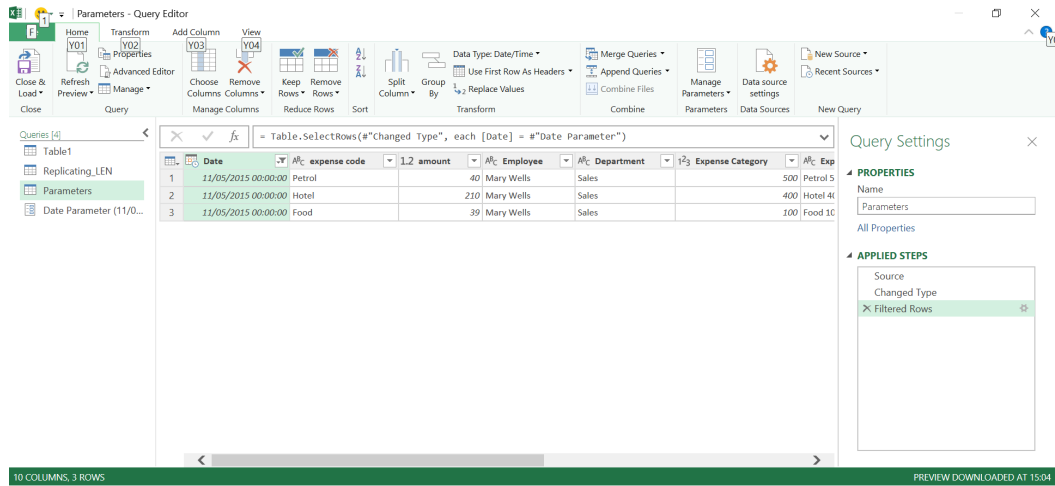
The current value has immediately been applied to the filter. In order to change the parameter, we can see it in the left-hand pane along with the current value. Another nice feature is that if we hover over the parameter, the description entered appears so we can be sure we have the right one. Let's click on the date parameter. Since we have chosen to allow any value, we can change it to another date:
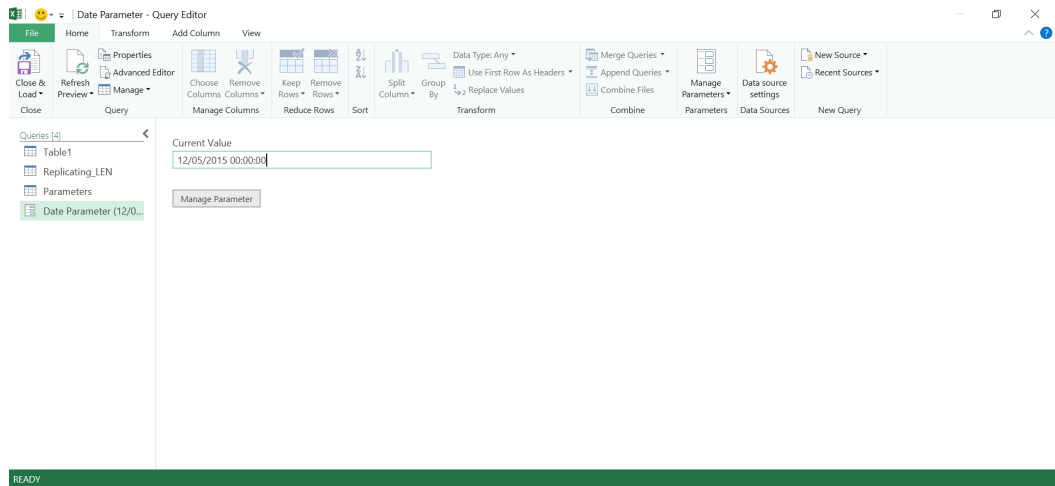
If we return to the *Parameters* query, our parameter change has been picked up by the filter and it remains in the same step 'Filtered Rows', since the step refers to my parameter:

= Table.SelectRows(#"Changed Type", each [Date] = #"Date Parameter")
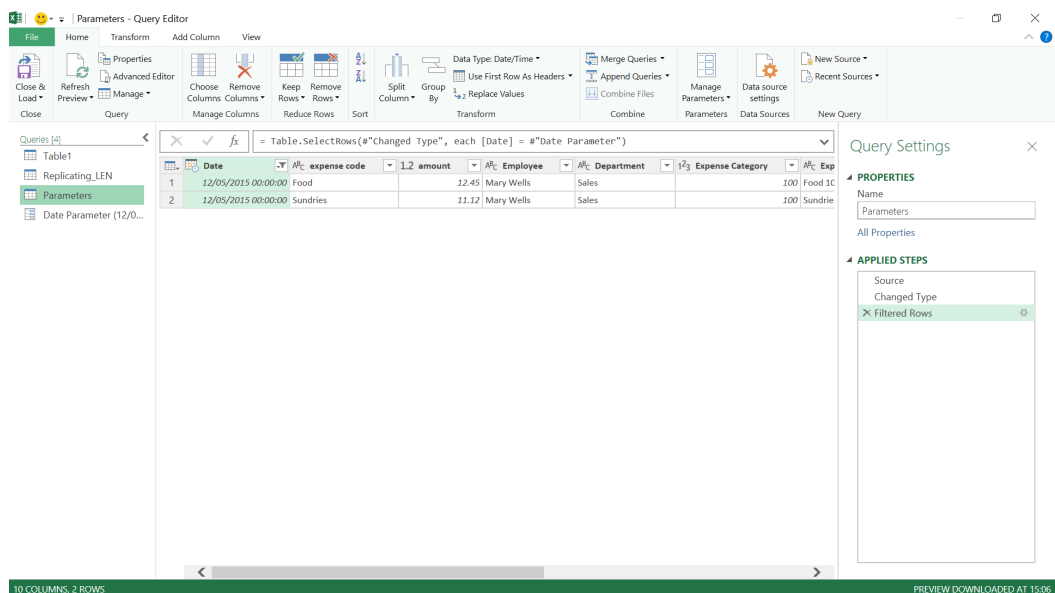
However, as we mentioned earlier, since we have allowed any value, we can pick dates which will return no data. To prevent this happening, we can use a list.

Let's return to the parameter screen by choosing the option 'Manage Parameter', and choose one of the list options, 'List of values'. We can enter a series of dates to create our list. Power Query will only allow us to add valid dates (Date/Time datatype) to this list:
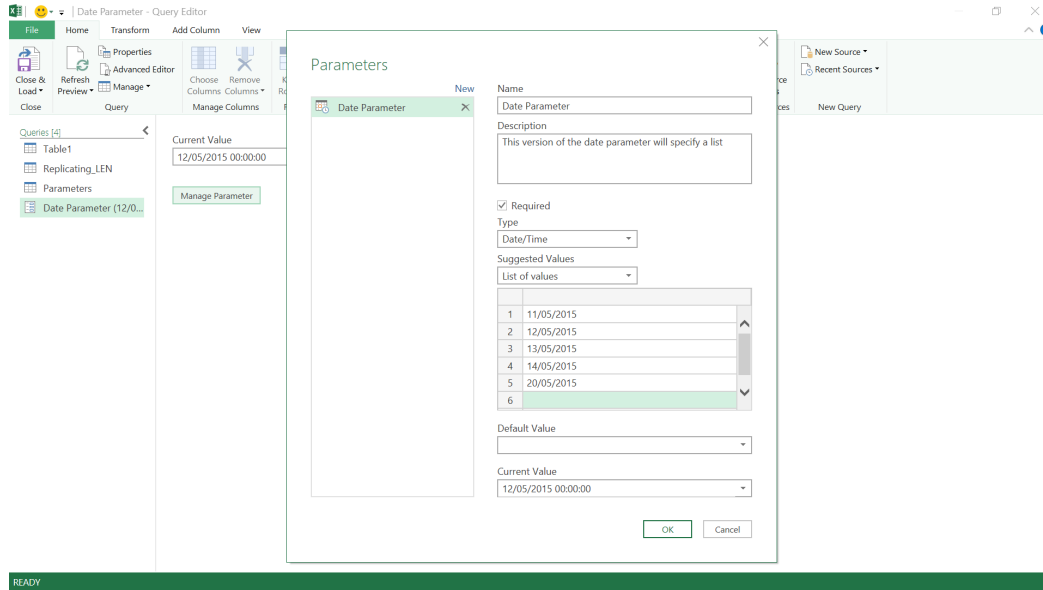
Check how the parameter entry works now.
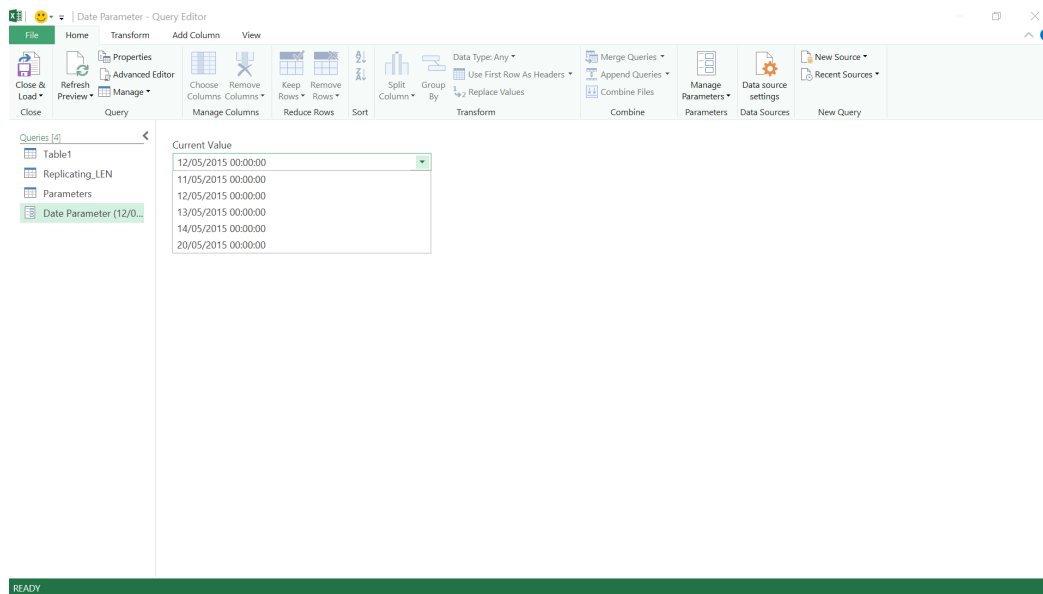
This time we can only pick from the dropdown, which means that we will always select at least one row when we change the parameter used in the filter.

More next month!

Power BI Desktop Update for August

The August release has another hotly anticipated feature from the Microsoft Business Application Summit, grouping. Grouping in reports, similar to PowerPoint's grouping, lets you easily organise groups of visuals to move and resize them as a set. This month's release also has some updates to Power BI's newest features, such as an icon set picker for the newly added icons conditional formatting and measure support for key influencers.

Here's the complete list of August updates:

Reporting

- Grouping visuals
- Filter pane auto-update
- Icon style picker for Icons conditional formatting
- Conditional formatting warnings

Analytics

- Key Influencers improvements
  - o Measure support
  - o General Availability

Visualisations

- xViz visualization suite by Visual BI
  - o Multiple Axis chart
  - o Marimekko chart
  - o Variance chart
  - o Horizon chart

Data connectivity

- Support for SAP HANA HDI Containers
- Edit SAP variables in Power BI Service (Preview)
- PostgreSQL DirectQuery (Beta)
- MarkLogic connector now Generally Available
- New Power Platform category within Get Data

Template apps

- Facebook Pages – Basic Analysis.

Let's take a look at each in turn.

Grouping visuals

Grouping allows you to group visuals, shapes, textboxes, images and buttons together on your report page similar to PowerPoint. Once objects are grouped together, you'll be able to move and resize them all together. This should make it a lot easier to work with reports with lots of objects layered on each page.
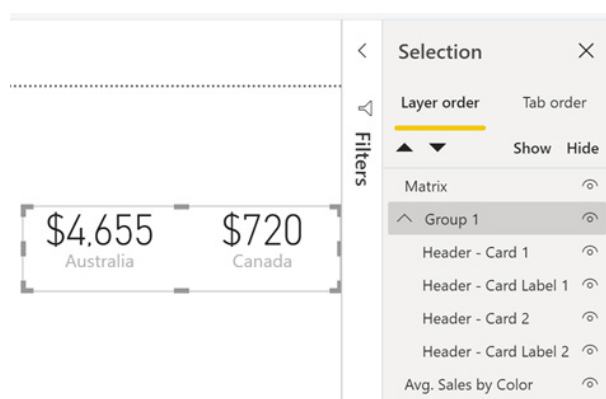
To group visuals, you can **CTRL + Click** them and right-click to select the Group option:
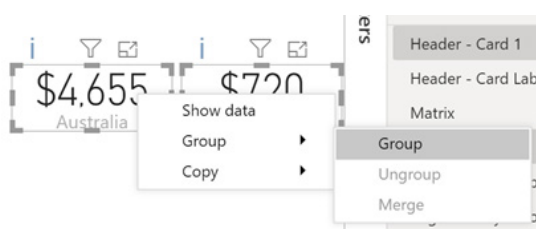
Right-clicking a group will give you an option to ungroup them later. Generally, by default, the behaviour mimics PowerPoint:

- Clicking on the empty space within the group will not select anything
- Clicking on a visual within a group will select the entire group on first click and the individual visual on second click
- Selecting a group and another object on the page and then selecting the Group option will create a nested group
- Use the selection pane to hide everything in the group at once (and bookmark it for you).

There are a few ways Microsoft has extended the feature as well, including:

- Use the Selection pane to drag and drop objects in and out of groups
- Selecting two groups and opening the context menu will give you an option to merge the groups into one group instead of nesting them
- Using the Formatting pane, you can add a background colour to a group. If a group has a background, it now has "substance" so you can click on the background of the group to select it.

Groups, since they aren't actual content on the report page, don't affect the tab order of your report page. Grouping is also only supported on reports using our new modern header, so if you have an older report still using the previous header, you'll need to upgrade it in the Options dialog to take advantage of this feature.

There are more plans afoot for grouping in the future as well. In a later update, different modes will be added to grouping that will change the way the visuals and the group's container interact with each other. For example, you'll eventually be able to create a group that scrolls.

Filter pane auto-update

Last month, Microsoft announced the General Availability of the new Filter pane, and now they are starting the process to transition current reports to the new pane as well. In this month's update, when you open Power BI Desktop for the first time, Power BI will show you a Filter pane auto-update dialog. Eventually, Power BI will auto-update all existing reports to the new Filter pane.
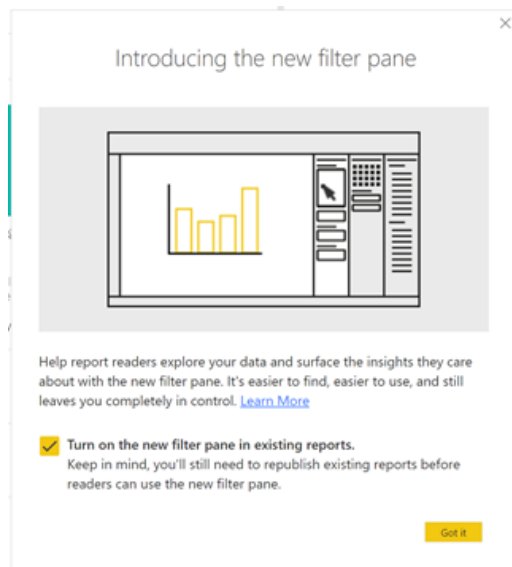
Through this dialog, you can opt into using the new Filter pane experience. Once you do this, anytime you open an old report it will show the new Filter pane automatically. If you save the report after opening it, the new pane will now persist, and once you publish the report again, it will show for you end users. If you're not ready to upgrade a specific report, you can turn off the new pane through the report level settings in the Options dialog.
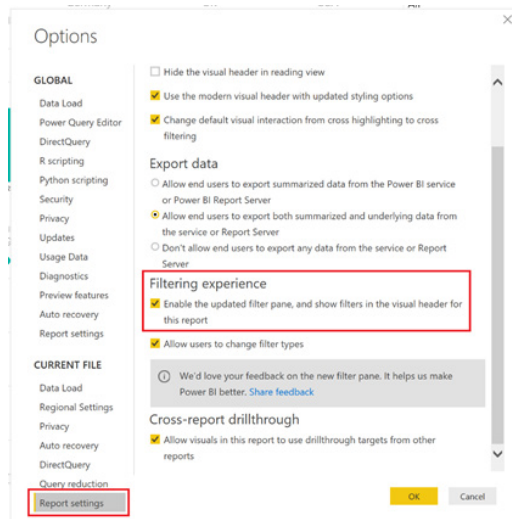
If you're not ready to switch to the new experience, you can close out the auto-update dialog or choose to opt out at this time. You'll still see the old pane for your reports, but when you're ready, you can opt in through the dialog the next time you open a report.

Icon style picker for Icons conditional formatting

Also last month, Power BI released a new form of conditional formatting, called Icons, that allows you to add icons next to values in your table or matrix based on custom business logic. This month, the usability of this feature is being improved by adding an icon set picker to the dialog.

This will give you a fast way to select a set of icons and rules to start with and then customise from there, which should save you some time manually selecting icons for each individual rule.
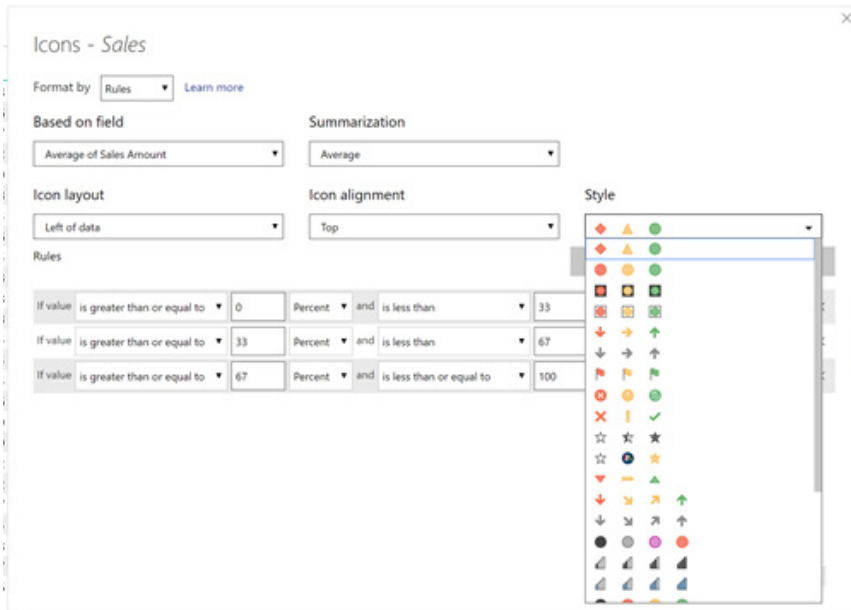
Conditional formatting warnings

Over the last several months, Power BI has expanded the number of places you can apply conditional formatting to include settings such as titles and font colours to name just a few. As mentioned before, the goal is to eventually support conditional formatting for most, if not all,

formatting properties. Given how prevalent conditional formatting will become with all these changes, Microsoft also wanted to make it easy to see errors in your conditional formatting configuration.
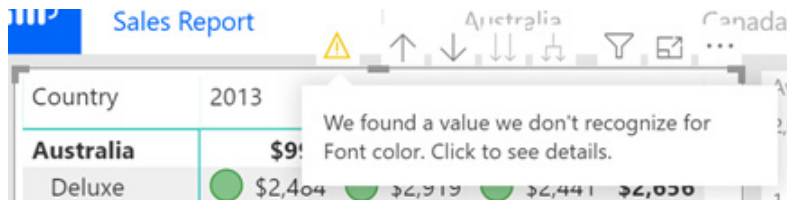
Now, Power BI will show a warning in the visual header and the Formatting pane if one of the following cases is true:

- The measure's data type is not what the formatting property expects (*e.g.* the measure returns a number, but the property is a string)
- The measure used to conditionally format was deleted
- The measure returns an unrecognized value (*e.g.* Rad instead of Red).

These warnings will only show up when editing a report; end users won't see the warnings when viewing the report.

Key Influencers improvements

MEASURE SUPPORT

You can now use measures in the Analyze bucket of the Key influencers visual. By default, Power BI will evaluate the measure based on the fields you include in the 'Explain by' bucket. There's also a new 'Expand by' bucket in the field well that can be used to refine the level that the

measure in the Analyze bucket is evaluated at. For example, if you want it to be evaluated over all your data, you can add an ID field to the 'Expand by' bucket.
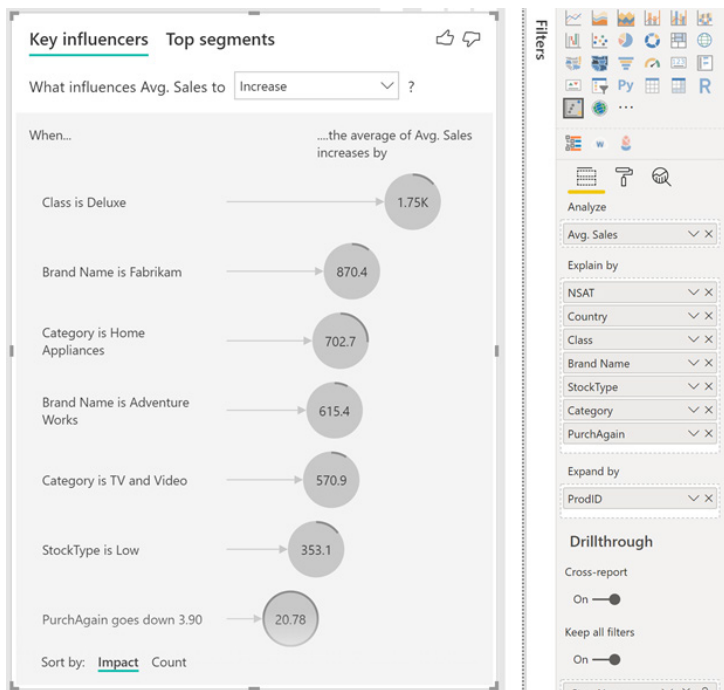
GENERAL AVAILABILITY

With measure support now added, Microsoft has now announced the General Availability of the Key influencers visual.

### *xViz visualization suite by Visual BI*

Visual BI, the makers of the ValQ value driver tree visual, are releasing a suite of new visuals available in AppSource called xViz. These in-app purchase visuals are highly customisable and are completely free up until 2000 data points. After that, they have a licensing plan in place that gives you access to all visuals within the suite for one price.

MULTIPLE AXIS CHART

Visual BI, the makers of the ValQ value driver tree visual, are releasing a suite of new visuals available in AppSource called xViz. These in-app purchase visuals are highly customisable and are completely free up until 2000 data points. After that, they have a licensing plan in place that gives you access to all visuals within the suite for one price.

To build out this visual, you'll start by placing the categorical axis field in the Axis bucket. From there, you'll place each measure you want to display in the value axis buckets. When you place several measures within the same value axis bucket, they'll share the axis and if you place them in different buckets, they will each have either own unique axis.
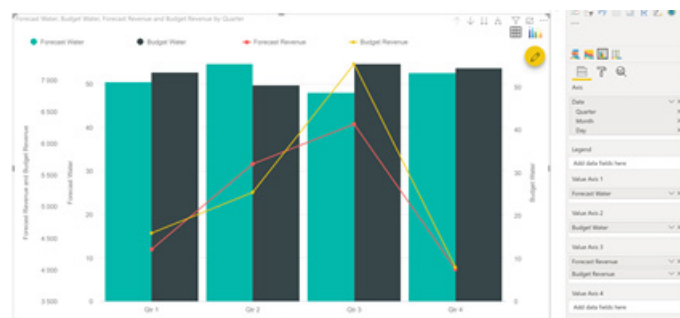
From there, there are numerous customisations you can do in the Formatting pane. You can customise the visual type using the 'Series Type' card, mixing and matching to get the visual layout you're looking for.
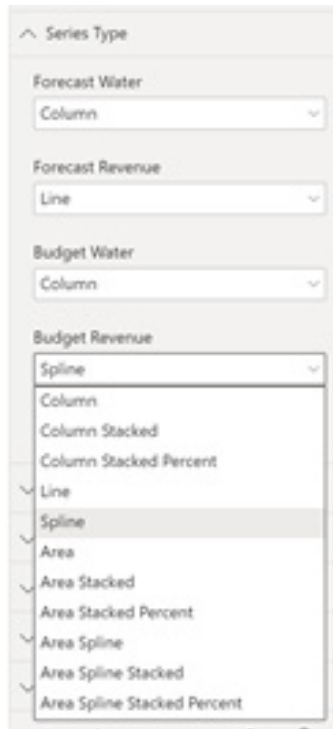
Some other unique options this visual has include:

- Error bars, horizontal and vertical bands and horizontal reference lines found in the Analytics card
- A utility menu that allows end users on the Power BI service to export PNGs, lasso select, and zoom (this menu is only supported in the service, not Power BI Desktop)
- Adding pattern fills using the 'Data Series Pattern' card to improve accessibility of your visuals
- Advanced formatting controls for each axis shown in the visual
- Custom text and URL footer through the Miscellaneous card

You can click on the grid icon on the top right of the visual to flip to a table view of the data and filter the rows through a search bar.

You can also click on the pencil icon to open a conditional formatting menu which allows you to write complex conditional formatting rules to apply to any series in the visual. The rules will automatically show in the legend as well (this can be turned off in the Formatting pane if needed).
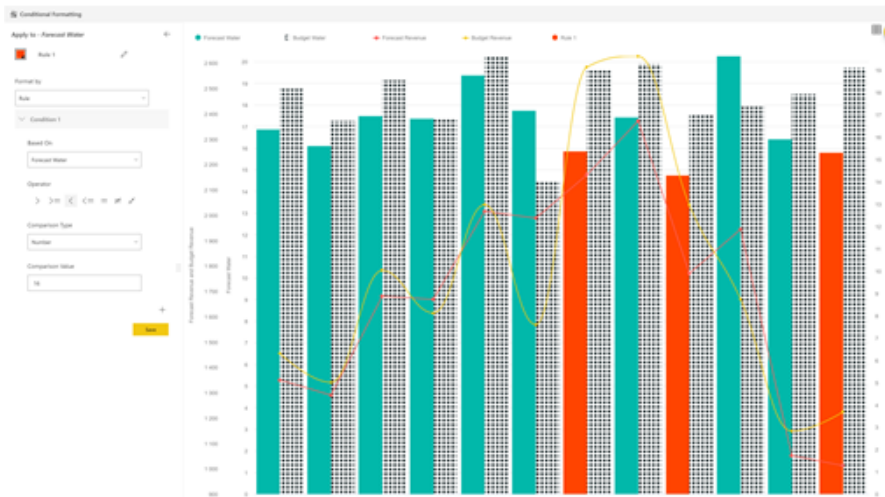
The conditional formatting and either be based on rules or on a colour (color) scale:
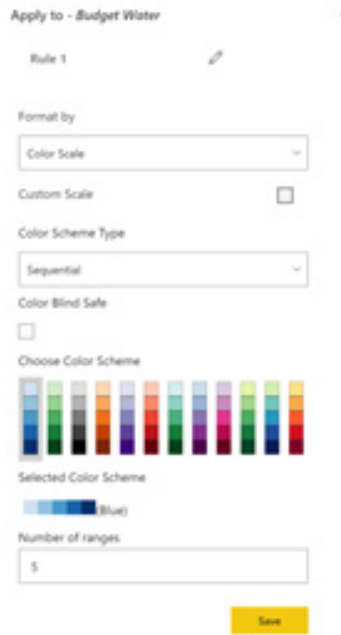
By combining all these features together, you can create complex and robust visuals to show your data however you need.
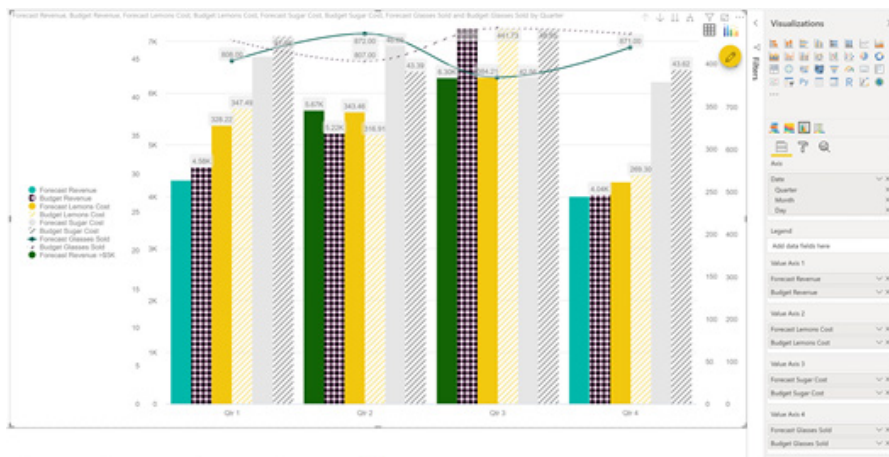
MARIMEKKO CHART

The Marimekko chart, the second xViz visual, lets you visualise categorical data over two scales, using both width and height, to show the relationship between them. You start off this visual with a categorical value you want to show on the value axis. From there, you can add a measure to define the height of each category and a different measure to set the width. You can also add a category to the legend to split the data further.
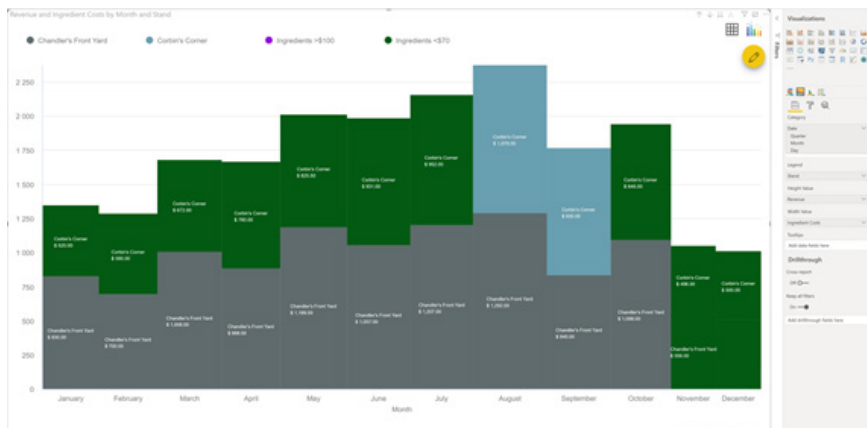
Since this visual is part of the xViz suite, it has similar formatting options to those covered for the Multi-axis chart *(above)*, including the conditional formatting options and advanced formatting control.

VARIANCE CHART

The variance chart, the third visual in the xViz suite, allows you to easily compare multiple metrics, such as forecast versus budget, and see the variance between them. For this visual, you need to add your categorical field, such as time, to the Category bucket in the field well. From there, you add the first value you want to compare in the Actual value bucket and the second to the Comparison value bucket. The visual will then show a comparison chart with both values, a variance chart, and a percentage variance chart.
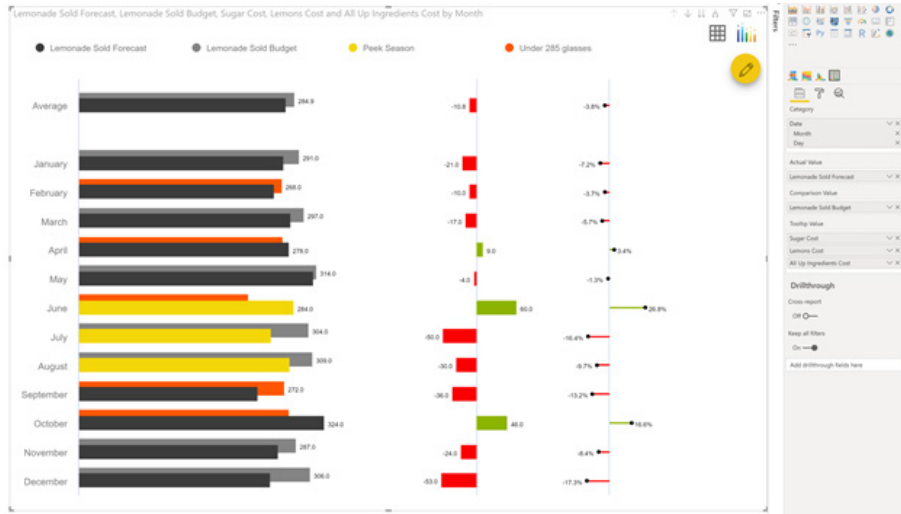
In addition to all the general formatting capabilities you get with all xViz visuals, such as conditional formatting, you also have a lot of controls specific to this visual type. For example, you can choose to customise which of these chart types you want to include in the visual and how much space they take up within the 'Chart Options' card of the Formatting pane.
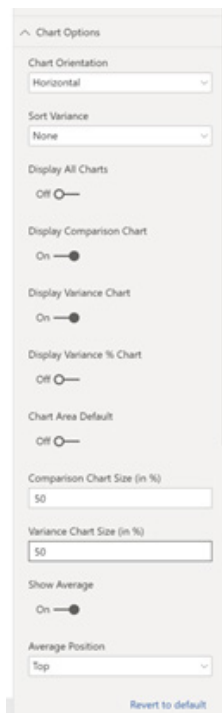
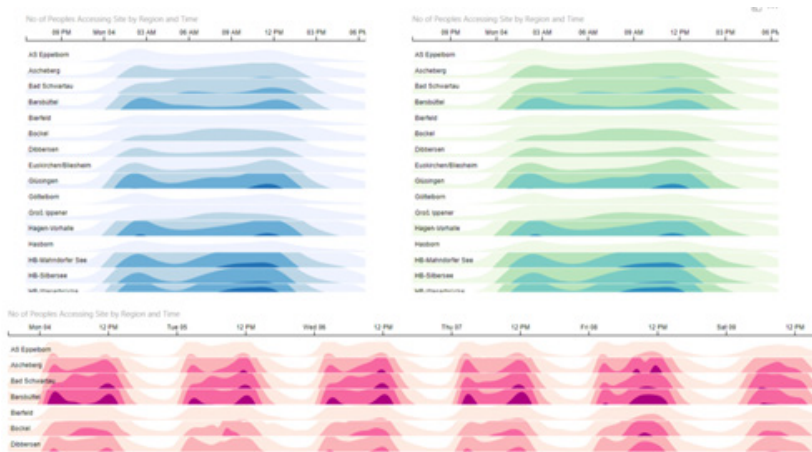You can switch the style of the variance percentage chart between oval and square through the Formatting pane as well.
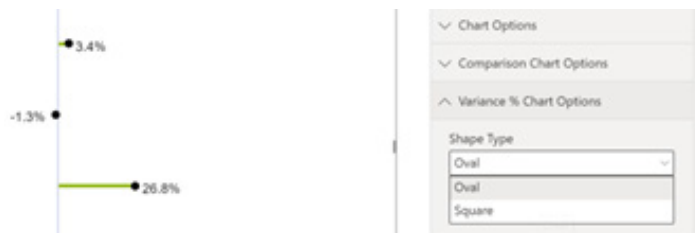
You can even change the chart type used for the comparison chart.
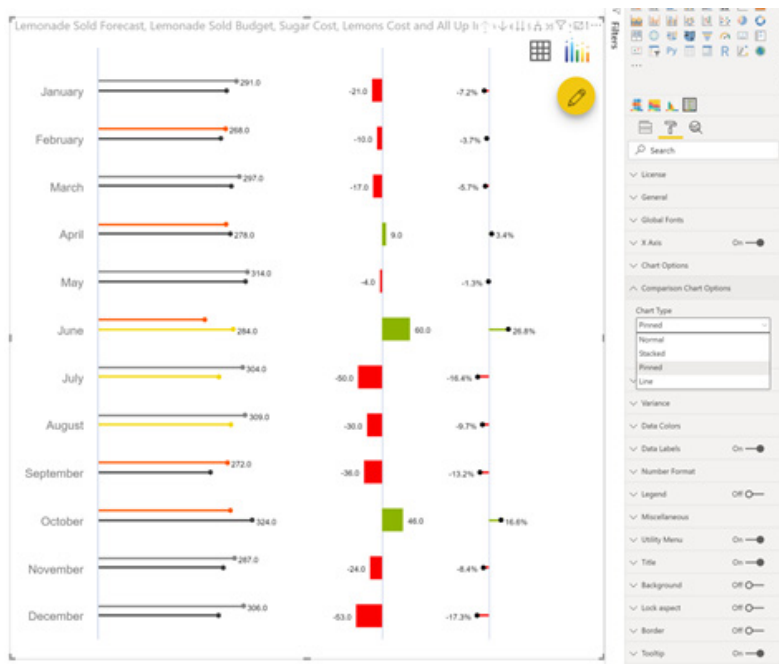
HORIZON CHART

Lastly, the Horizon chart, which was originally issued last September, has been upgraded and integrated into the xViz suite.

Support for SAP HANA HDI Containers

The SAP HANA connector has been enhanced in this month's release to enable consumption of views under HANA Deployment Infrastructure ("HDI") Containers. Views within HDI Containers are seamlessly showed within the Navigator dialog when using the SAP HANA connector, allowing you to consume them with either Import or DirectQuery mode and operate over them just like with any other view within SAP HANA.

Edit SAP variables in Power BI Service (Preview)

When using SAP Business Warehouse or SAP HANA with DirectQuery, you can now allow end users to edit SAP variables in the Power BI service for Premium workspaces.

It should be noted that this is a Preview feature that requires the new filter experience and an SSO-enabled gateway.

PostgreSQL DirectQuery (Beta)

Another connector enhancement this month provides DirectQuery support within the PostgreSQL connector, enabling you to build reports on top of their PostgreSQL databases without the need to import data.

When launching the PostgreSQL connector, you can now specify DirectQuery as your connectivity mode choice within the connector dialog.
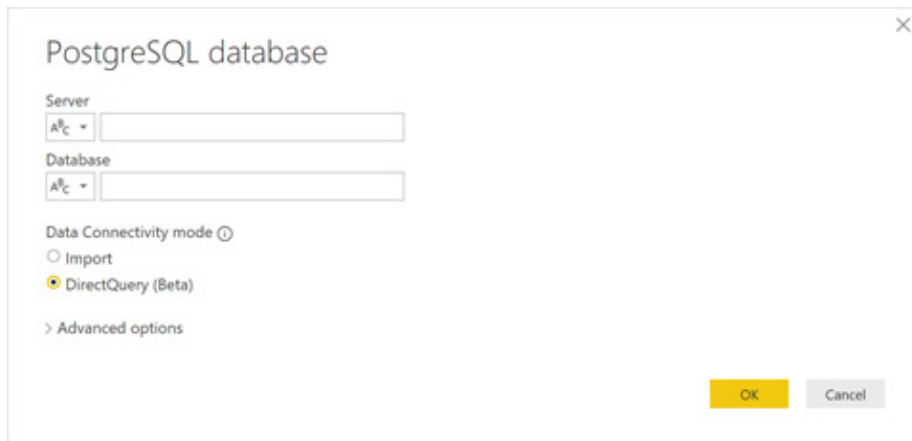
MarkLogic connector now Generally Available

The MarkLogic connector is now Generally Available. It may be found within the Database category of the 'Get Data' dialog.
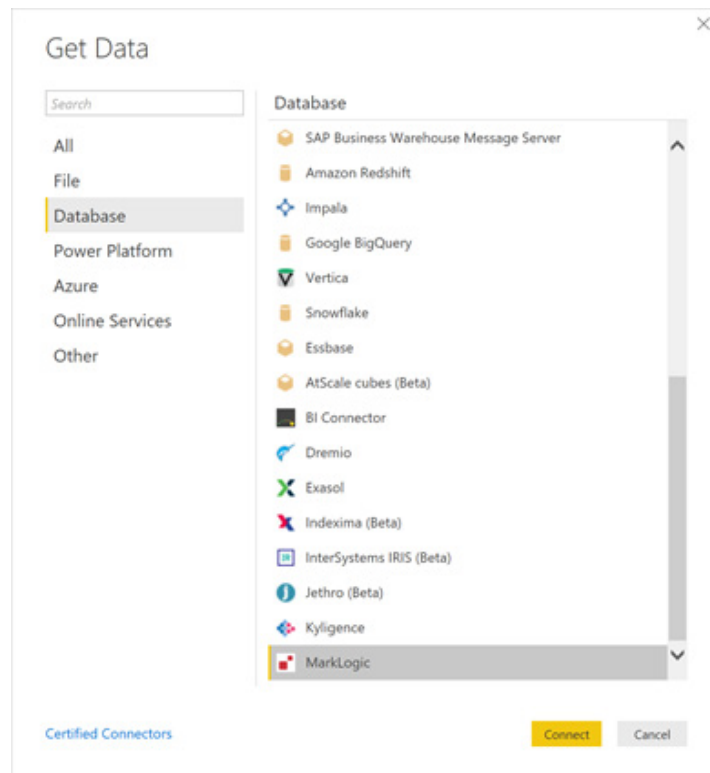
### New Power Platform category within Get Data

With this month's release, we're renaming the Power BI category under Get Data to **Power Platform**, and bringing the Common Data Service connector into this category, in addition to the Power BI datasets and Power BI dataflows connectors.

According to Microsoft, this change also lays the foundation for several other enhancements coming to Power BI that will leverage other Power Platform capabilities provided by PowerApps, Flow, and the Common Data Service. Stay tuned!

Facebook Pages – Basic Analysis

Template apps are a great way to provide Power BI users with installable, ready to use, actionable – quality applications. These templates contain reports, dashboards and data models that can connect to the user's data, creating immediate value and action.

The first of these is the Facebook Pages – Basic Analysis template app by DataChant, which provides marketing professionals the simplest way to analyse any Facebook page in Power BI without being the page administrator.
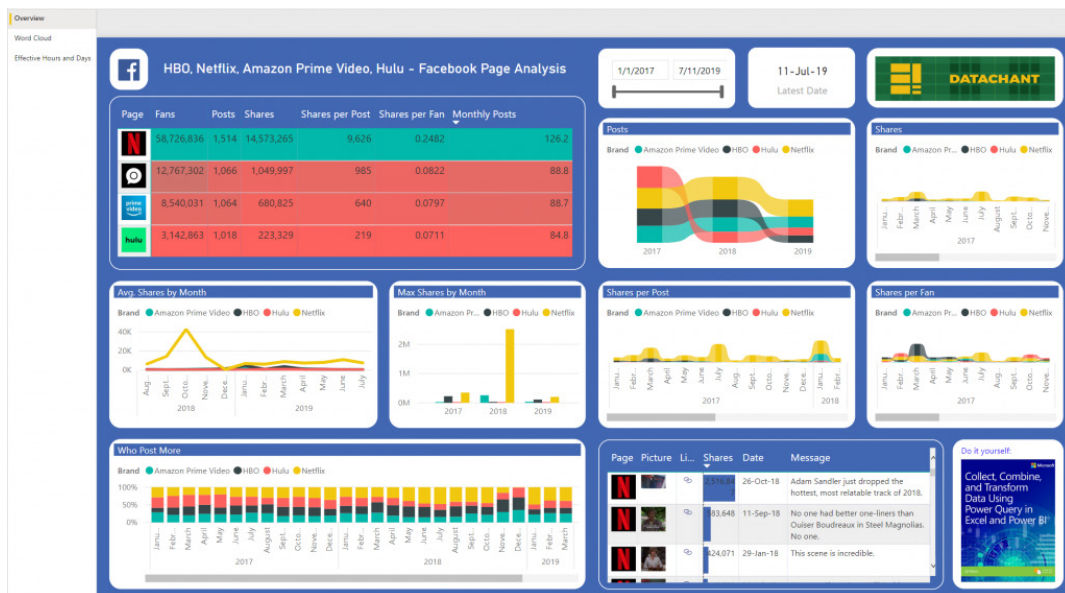
You can install the app right in the Power BI service and set its parameters to import data about up to four Facebook pages.

Once you complete the short set up flow, you'll be given a dashboard and report that shows how each page performs over time based on their number of posts and shares. Even if you're not a social media analyst, you can use this template app as a great demo tool to showcase Power BI capabilities. The dashboard and report are now yours, so you can customise and share as much as you want.

More next month we're sure!

Latest Updates for Power BI Service and Mobile

And it keeps coming! Here's the latest updates for Power BI Service and Mobile:

- Support for non-premium datasets for Paginated reports
- General Availability of Power BI aggregations
- Update for On-premises data gateway
- Send refresh failure notifications to others
- Power BI Premium capacity testing tool
- Service availability notifications (Preview)
- New Look (Preview) in the Power BI service
- New Power BI region in South Africa
- Report page ToolTips on Power BI mobile
- Roadmap updates

## Support for non-premium datasets for Paginated reports

This latest update to Power BI Report Builder includes, for the first time, the ability to connect to your Power BI datasets, regardless of them being backed by premium capacity. It means any Power BI Pro user can connect to and author, render and export reports locally against any Power BI datasets with the Power BI Report Builder.
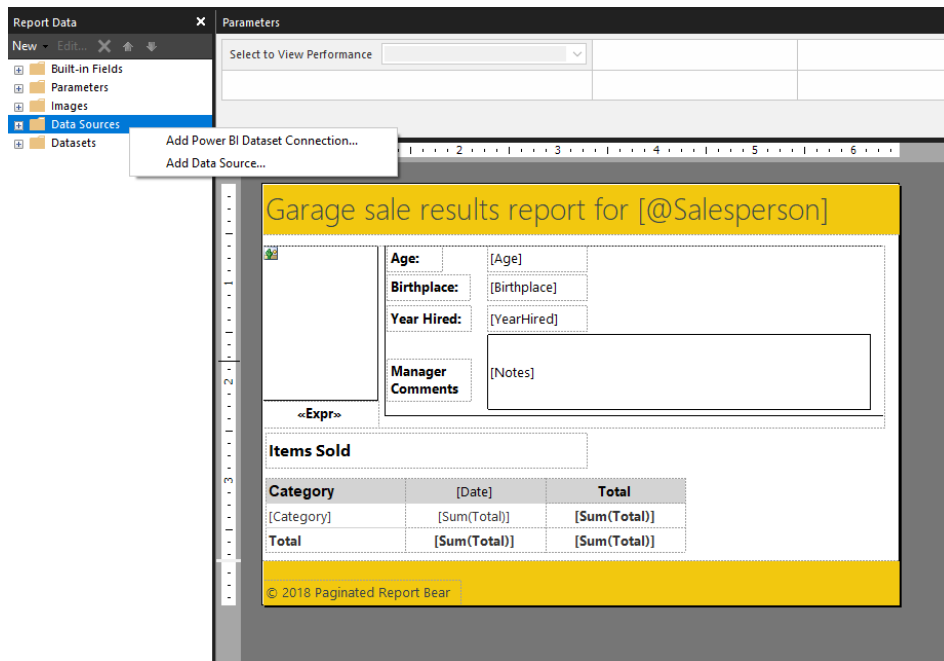
General Availability of Power BI aggregations

Power BI aggregations are now Generally Available – one of the biggest scalability features Microsoft has in its BI stack. Aggregations empower organisations to enable every user to perform their own analysis atop full fidelity, petabyte scale datasets. At the same time, those organisations now have better control over their BI architecture, striking a better balance between cost, performance and data fidelity. In addition, Microsoft announced that aggregations can be used for enterprise-grade semantic models in large organisations with fine-grain security requirements by leveraging Power BI's row level security feature (RLS).
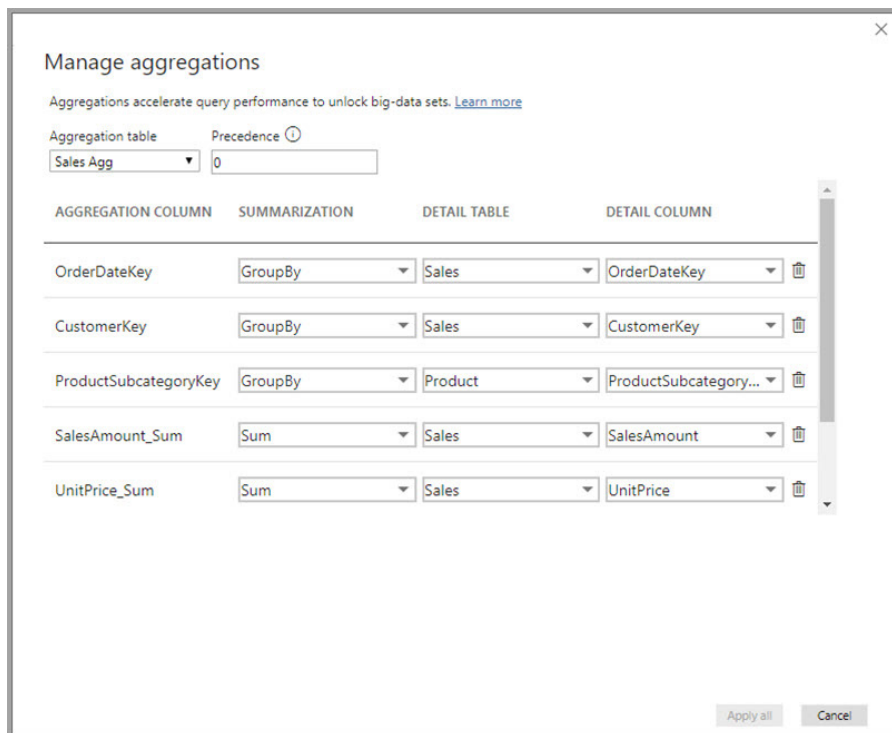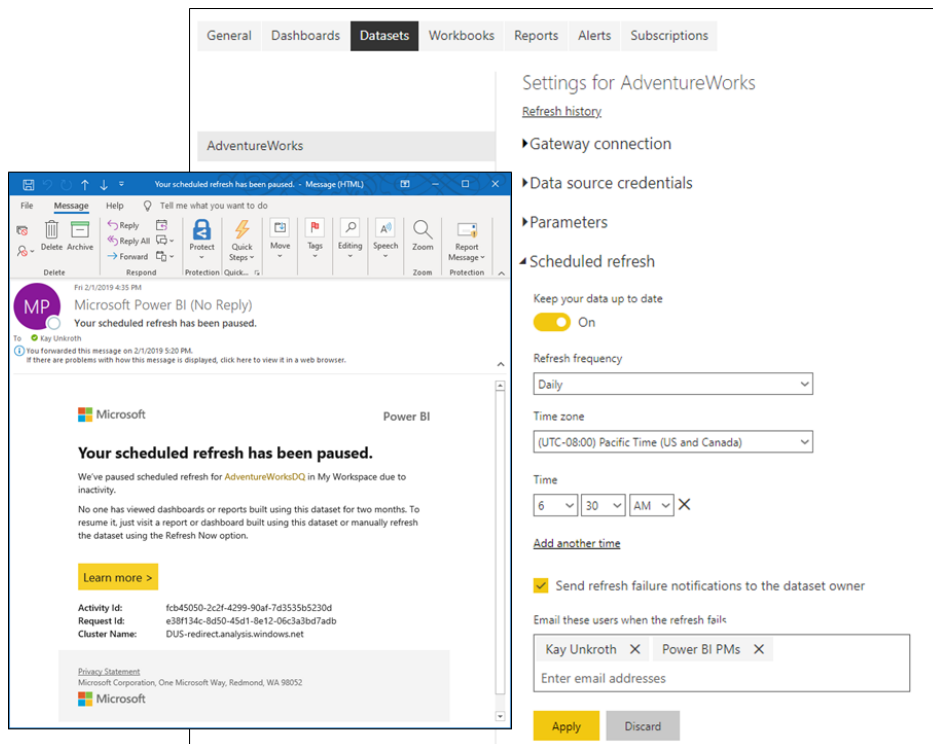

Update for On-premises data gateway

There's an updated version of Power BI On-premises data gateway. The latest version has an updated mashup engine, update to the .Net Framework, and a bug fix to the encrypted connection.

Send refresh failure notifications to others

There's also a new feature under Scheduled Refresh where you can specify additional users and email-enabled groups to receive the refresh failure notifications in addition to the dataset owner. Sending refresh notifications to others can deliver significant value, enabling you to monitor your scheduled refreshes more reliably and more effortlessly than before.


Power BI Premium capacity testing tool

To help you understand how much user load your Power BI Premium capacities can handle when being used, Microsoft has introduced a new load testing tool that will help you create automated load tests against your capacities, and lets you choose which reports to test and how many concurrent users to simulate. The tool will generate load on a capacity of choice by continuously rendering each report with new filter values (to prevent unrealistically good performance due to report caching) until the token required for authenticating the tool against the service expires.

Service availability notifications (Preview)

To help you stay on top of Power BI service availability, this update launched a new feature that you can opt in for, to receive email notifications when there is a service disruption or degradation. While Power BI's 99.9% SLA makes these occurrences rare, Microsoft added this functionality to ensure that you are covered even for those rare events. As a tenant administrator, you can create a security group with all the users that will should these notifications.

The user interface for the Power BI service has been refreshed. It introduced an opt-in 'new look' experience that's simpler and builds on customers' familiarity with Microsoft products. This initial release has updates to the report viewing experience, and there's more to come in the upcoming months.
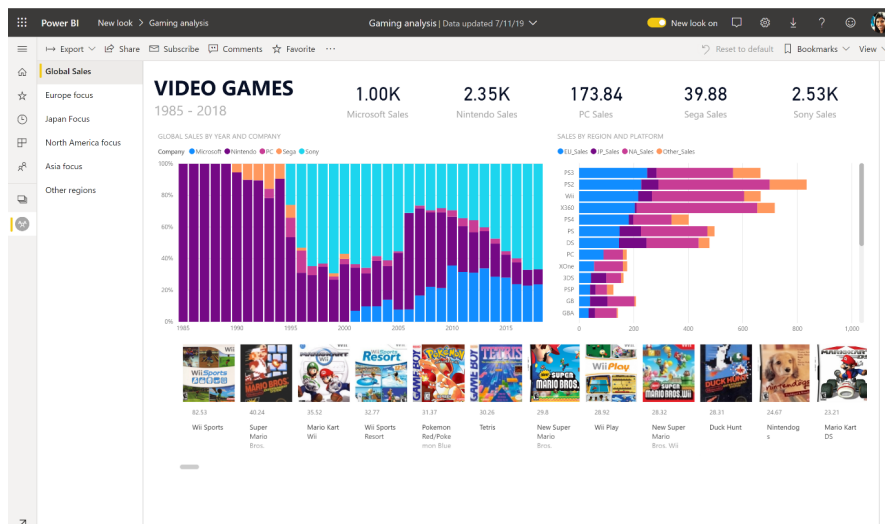
New Power BI region in South Africa

Power BI has been expanded to add its first available region in Africa, the South Africa North region, located in Johannesburg. With the addition of this new region, Power BI is now available in 19 regions in Microsoft's global public cloud, in addition to six regions in their three national clouds, Power BI for US Government, Power BI German, and Power BI in China (operated by 21Vianet). You may immediately leverage these by deploying your Power BI Multi-Geo Premium capacity.

Report page ToolTips on Power BI mobile

Mobile now has ToolTips. Now, you just need to simply press and hold on a visual that has a report tooltip tied to it and the tooltip will be shown.
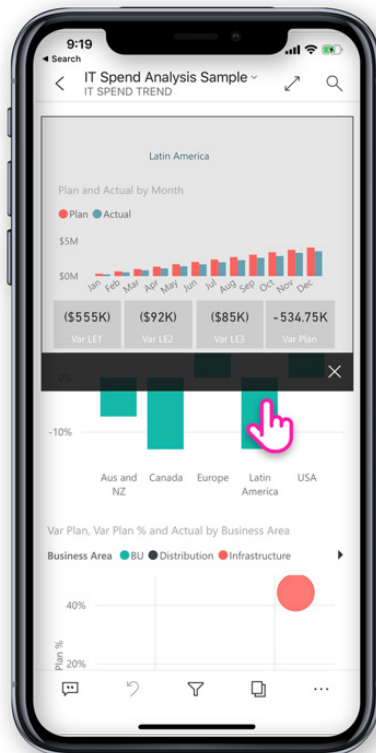
Roadmap updates

Here's the latest disclosures:

- Data protection
- Datasets larger than 10 GB in Power BI Premium
- Mobile Home (Android, iOS)
- Unattended app installation
- In-place app update.

Stay tuned for the next update.

Power BI MVP Book

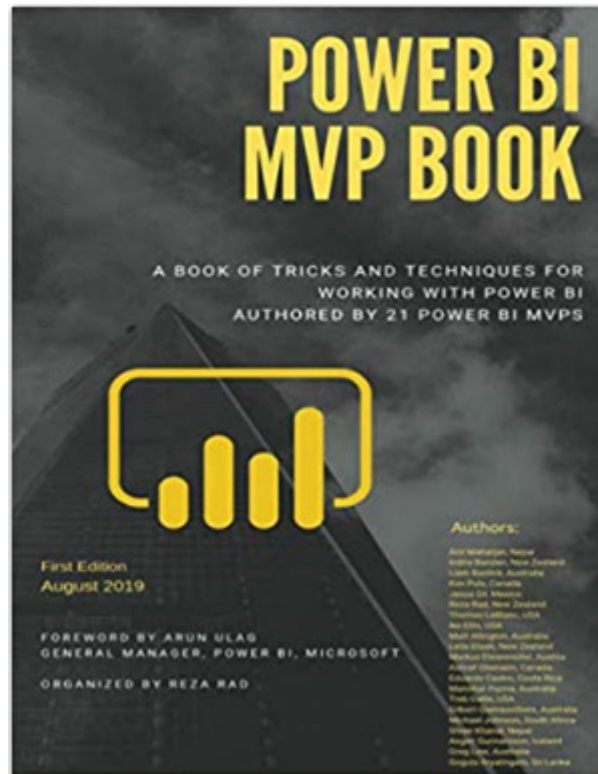There's a new Power BI book out – but it's not like a normal book used to learn Power BI. This book is a selection of articles from 21 Power BI MVPs, including our very own **Liam Bastick**. Each chapter focuses on a particular subject instead, rather than provide a "cradle to grave" reference to the software. It's about tricks and techniques instead.

Topics covered include: AI, architecture, data modelling, data preparation, DAX, enterprise issues and Integration. The book is organised such that you may pick and choose any chapter in any order.

It's available from Amazon, amongst other sellers, and we take great pride in noting that on Amazon Australia, the Kindle edition is currently the #1 Best Seller in Data Modelling and Design!!

Power BI MVP Book: A book of tricks and techniques for working with Power BI Kindle Edition

by Reza Rad (Author), Anil Maharjan (Author), Indira Bandari (Author), Liam Bastick (Author), Ken Puls (Author), & 16 more

Be the first to review this item

#1 Best Seller in Data Modelling & Design

> See all formats and editions

Kindle
$0.00 kindleunlimited

This title and over 1 million more available with Kindle Unlimited
$11.99 to buy

This Power BI MVP book is not like a normal book used to learn Power BI. This book is a selection of articles from 21 Power BI MVPs. Each chapter focuses on a particular subject.

Topics covered include: AI, architecture, data modeling, data preparation, DAX, enterprise issues and Integration.
‹ Read more

The A to Z of Excel Functions: DECIMAL

This function converts a text representation of a number in a given base into a decimal number (base 10).

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

DECIMAL(text, radix)

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

- **text:** this is required
- **radix:** this is also required and must be an integer.

It should be further noted that:

- the string length of **text** must be less than or equal to 255 characters
- the **text** argument can be any combination of alpha-numeric characters that are valid for the **radix**, and is not case sensitive
- Excel supports a **text** argument greater than or equal to 0 and less than 2^{53}. A **text** argument that resolves to a number greater than 2^{53} may result in a loss of precision
- **radix** must be greater than or equal to 2 (binary, or base 2) and less than or equal to 36 (base 36)
- a **radix** greater than 10 use the numeric values 0-9 and the letters A-Z as needed. For example, base 16 (hexadecimal) uses 0-9 and A-F, and base 36 uses 0-9 and A-Z
- if either argument is outside its constraints, **DECIMAL** may return the *#NUM!* or *#VALUE!* error value.

Please see our example below:

	A	B	C	D
1	**Formula**	**Description**	**Result**	**How It Works**
2	*=DECIMAL("FF",16)*	Converts the hexadecimal (base 16) value FF to its equivalent decimal (base 10) value (255).	255	"F" is in position 15 in the base 16 number system. Since all number systems start with 0 (zero), the 16th character in the hexadecimal system will be in the 15th position.
3	*=HEX2DEC("FF")*	The **HEX2DEC** function in cell **C3** verifies the above result.	255	=(15*(16^1))+(15*(16^0))
4	*=DECIMAL(111,2)*	Converts the binary (base 2) value 111 to its equivalent decimal (base 10) value (7).	7	1 is in position 1 in the base 2 number system.
5	*=BIN2DEC(111)*	The **BIN2DEC** function in cell **C5** verifies the above result.	7	=(1*(2^2))+(1*(2^1))+(1*(2^0))
6	*=DECIMAL("Liam",36)*	Converts the hexadecimal (base 16) value FF to its equivalent decimal (base 10) value (255).	1,003,486	"L" is in position 21 in the base 36 number system, "i" is in position 18, "a" is in position 10 and "m" is in position 22.
7				=(21*(36^3))+(18*(36^2))+(10*(36^1))+(22*(36^0))

The A to Z of Excel Functions: DEGREES

This function will get you qualified, provided you come at it from the right angle. This function converts radians into degrees.
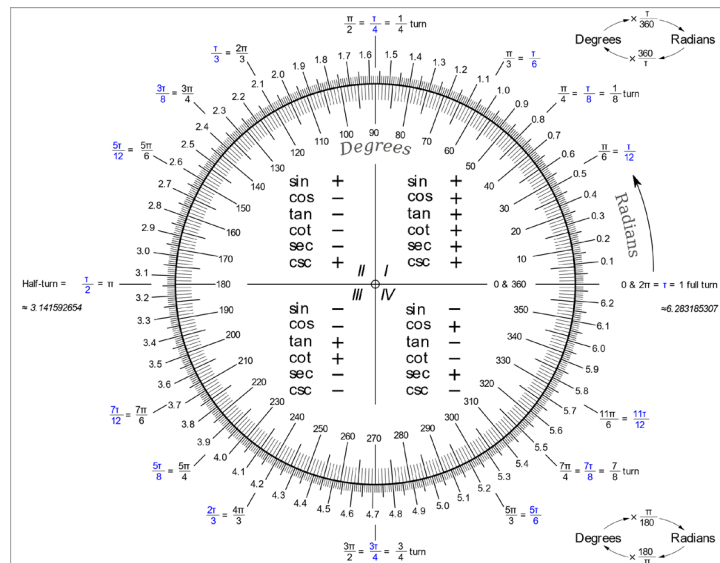
The **DEGREES** function employs the following syntax to operate:

DEGREES(angle)

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

- **angle:** this is required and represents the **angle** in radians that you wish to convert.

Please see our final example below:

	A	B	C
1	**Formula**	**Description**	**Result**
2	*=DEGREES(PI())*	Degrees of pi radians	180

More Excel Functions next month…

Upcoming SumProduct Training Courses

Location	Course	Date	Duration
Melbourne	Excel Tips and Tricks	18 Sep 2019	1 Day
Sydney	Excel Tips and Tricks	23 Sep 2019	1 Day
Sydney	Financial Modelling	24 - 25 Sep 2019	2 Days
Sydney	Power Pivot, Power Query and Power BI	7 - 9 Oct 2019	3 Days
Melbourne	Financial Modelling	14 - 15 Oct 2019	2 Days
Melbourne	Power Pivot, Power Query and Power BI	22 - 24 Oct 2019	3 Days
Sydney	Excel Tips and Tricks	4 Nov 2019	1 Day
Sydney	Financial Modelling	5 - 6 Nov 2019	2 Days
Sydney	Power Pivot, Power Query and Power BI	11 - 13 Nov 2019	3 Days
Melbourne	Excel Tips and Tricks	25 Nov 2019	1 Day
Melbourne	Financial Modelling	26 - 27 Nov 2019	2 Days
Melbourne	Power Pivot, Power Query and Power BI	11 - 13 Dec 2019	3 Days
Sydney	Power Pivot, Power Query and Power BI	9 - 11 Dec 2019	3 Days
Sydney	Excel Tips and Tricks	16 Dec 2019	1 Day
Sydney	Financial Modelling	17 - 18 Dec 2019	2 Days

Key Strokes

Each newsletter, we'd like to introduce you to useful keystrokes you may or may not be aware of. This month, we thought we would provide a reminder of the function keys:

Keystroke	What it does
F1	Help
F2	Toggle Select, Edit and Enter / Point modes
F3	Paste names
F4	Standard mode: Redo / Edit (F2) mode): toggle $ references
F5	Go To / Refresh File List
F6	Next pane
F7	Check spelling
F8	Extend Selection mode
F9	Calculate now
F10	Activate menus
F11	Insert chart on new worksheet
F12	Save As

There are over 540 keyboard shortcuts in Excel. For a comprehensive list, please download our Excel file a 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.