

SumProduct

NEWSLETTER #129 - August 2023

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



It's mid-year and it looks like the re-runs have already started...

But it's not true though! SumProduct celebrates that two of its gang retain their MVP status and we celebrate with the announcement of an old favourite: Excel Virtually Global! And to keep the "Oldies But Goodies" theme, we even revisit one of Excel's least understood functions, **AGGREGATE**.

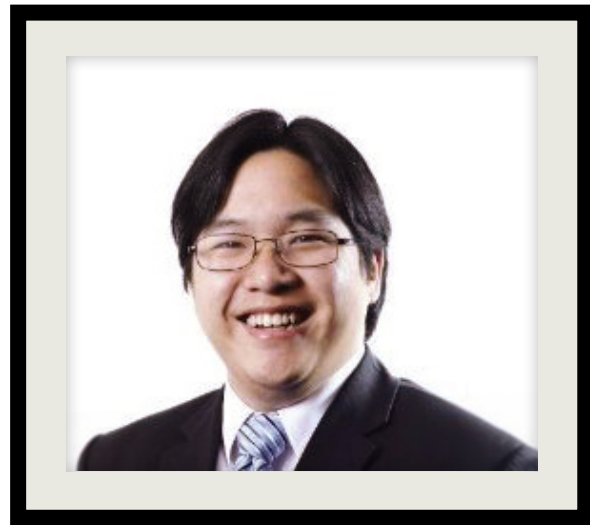
This month, we also include all the usual newsletter goodies such as the Beat the Boredom Challenge, Charts & Dashboards, Visual Basics, Power Pivot Principles, Power Query Pointers, Excel Updates, literally the Deduce our A to Z of Keyboard Shortcuts and **LOG** the A to Z of Excel functions.

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

Liam Bastick, Managing Director, SumProduct



MVP Renewal



SumProduct is pleased to announce that two of our Directors, **Liam Bastick** and **Tim Heng**, have been re-awarded Microsoft's Most Valuable Professional (MVP) award for Excel for 2023-24. Not unluckily for Liam, he receives his 13th award, whilst Tim forges ahead with his sixth. This award recognises exceptional technical community leaders from around the world who voluntarily share their high quality, real world expertise with others. Microsoft MVPs are a highly select group of experts representing technology's best and brightest who share a deep commitment to community and a willingness to help others.

Worldwide, there are over 100 million participants in technical communities; of these participants, there are c.4,000 active Microsoft

MVPs. In Excel, we believe there are approximately 80 that have received this award.

Microsoft's MVP Award evaluates technical expertise and voluntary community contributions for the past year, considering the quality, quantity and level of impact of contributions. It's a difficult award to attain and just as difficult to retain.

At SumProduct, you can rely on our experience and willingness to help - simply drop us a line at contact@sumproduct.com.





EXCEL VIRTUALLY GLOBAL 2023

We don't have all the details yet – but it's time to make an announcement it's "coming soon"!!

The MVPs and other experts are back to passionately share their knowledge with the community. They are always on the "bleeding edge" and have an unstoppable urge to get their hands on new, exciting technologies.

How much would you be prepared to pay for the likes of Excel (and other) MVPs, Microsoft staff and other industry experts to present on a variety of Excel topics **in your house**?

Well, it's an option. For the fourth year running, we are remaining virtual (maybe next year...). And it still remains for charity.

This virtual conference presents Excel, Data Platform and PowerPoint MVPs and other experts, together with other acknowledged experts from Microsoft and / or around the globe to present, answer questions and demonstrate the future of Excel and how it will make your life easier, personally and professionally. And it's a big year this year, with Excel, ChatGPT, Fabric, Copilot, Power Apps and financial modelling in Power BI all staking their claims in the modelling world.

Topics and presenters have not yet been agreed, but you know the drill! We'll round a few up and get them to entertain you. Dates will

be announced SOON, but we anticipate sessions to begin in the week commencing Monday 9 October 2023.

Each session (including Q&A) will last no more than an hour and topics cover all expertise levels, from novice to expert. Most presenters are well known in their spheres, and have written blogs, books and articles and / or present video sessions.

Most sessions will be recorded so you may watch them later with downloads aplenty – there are no medals for staying up to watch the entire event live! That's just as well, as we plan to beat last year's marathon (no guarantees), which lasted the best part of two [2] days. 😊

From your own favourite chair, bring a laptop, an inquisitive mind and your sense of humour. Be prepared to learn heaps. And remember, it's not just value for money, we will be asking for donations to charity this time around – all profits will be donated to a global COVID-19 research charity.

More details will be available in the News section of our website soon: www.sumproduct.com/news.

Hopefully, we'll see you there!

AGGREGATE: Possibly the most complicated Excel function ever..?

AGGREGATE() first appeared in Excel 2010. And ever since, it has caused constant consternation amongst the modelling fraternity. Never mind ChatGPT, **LET** and **LAMBDA**, some of the most common questions we get here at SumProduct HQ emanate from the wonderful **AGGREGATE** function. Therefore, we thought we would revisit it this month, as it often seems so *neglected*.

For those who desire greater sesquipedalian loquaciousness (look it up!), its syntax may give some comfort as it has two forms:

- Reference: =**AGGREGATE**(function_number, options, ref1, [ref2],...)
- Array: =**AGGREGATE**(function_number, options, array, [optional_argument])

where:

- **function_number** denotes function that you wish to use. Similar to **SUBTOTAL**, **function_number** allocates integer values to various Excel functions:

Comparison		
SUBTOTAL	AGGREGATE	Function
1	1	AVERAGE
2	2	COUNT
3	3	COUNTA
4	4	MAX
5	5	MIN
6	6	PRODUCT
7	7	STDEV.S
8	8	STDEV.P
9	9	SUM
10	10	VAR.S
11	11	VAR.P
	12	MEDIAN
	13	MODE.SNGL
	14	LARGE
	15	SMALL
	16	PERCENTILE.INC
	17	QUARTILE.INC
	18	PERCENTILE.EXC
	19	QUARTILE.EXC

- **options** specifies which values may be ignored when applying the chosen function to the range. If the **options** parameter is omitted, the **AGGREGATE** function assumes that **options** is set to zero [0]. The argument **options** can take any of the following values:

Option	Behaviour
0	Ignore nested SUBTOTAL and AGGREGATE functions
1	Ignore hidden rows, nested SUBTOTAL and AGGREGATE functions
2	Ignore error values, nested SUBTOTAL and AGGREGATE functions
3	Ignore hidden rows, error values, nested SUBTOTAL and AGGREGATE functions
4	Ignore nothing
5	Ignore hidden rows
6	Ignore error values
7	Ignore hidden rows and error values

- **ref1** is the first numeric argument for the function when using the reference syntax
- **ref2,...** is optional. Numerical arguments may number two [2] through 253 for the function when using the reference syntax
- **avrray** is an array, array formula or reference to a range of cells when using the array syntax
- **optional_argument** is a second argument required if using the **LARGE**, **SMALL**, **PERCENTILE.INC**, **QUARTILE.INC**, **PERCENTILE.EXC** or **QUARTILE.EXC** when using the array syntax, viz.

Function	Optional Argument Required
LARGE	LARGE(Array,k)
SMALL	SMALL(Array,k)
PERCENTILE.INC	PERCENTILE.INC(Array,k)
QUARTILE.INC	QUARTILE.INC(Array,Quart)
PERCENTILE.EXC	PERCENTILE.EXC(Array,k)
QUARTILE.EXC	QUARTILE.EXC(Array,Quart)

AGGREGATE is analogous to an extension of the **SUBTOTAL** function insofar that it uses the same **function_number** arguments, simply adding another eight [8]. **SUBTOTAL** allows users to use the 11 functions including / excluding hidden rows which results in 22 combinations. **AGGREGATE** goes further and takes the 19 functions and allows for eight

alternatives for each, which results in 152 combinations – and that’s not even considering the reference or array syntax approaches!

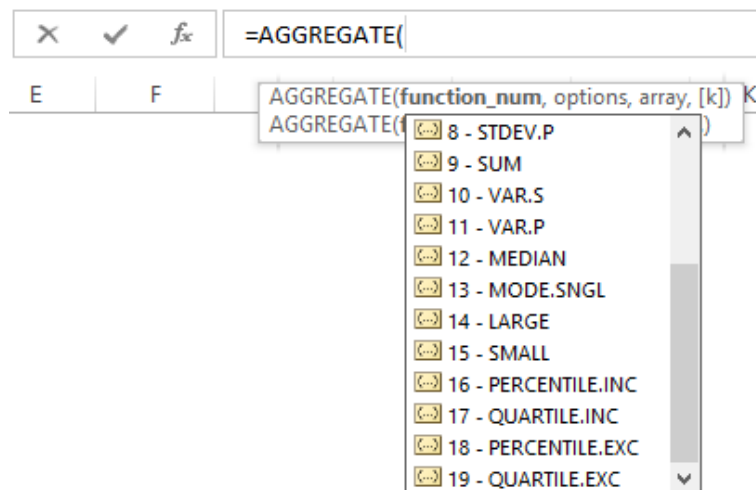
It just all sounds, well, tremendously complicated.

In practice, it's not that bad. This is because since this function was created for Excel 2010 and later, screen tips will appear as you type in order to nudge you in the right direction. For example, let's say you wanted the third largest number in the following list:

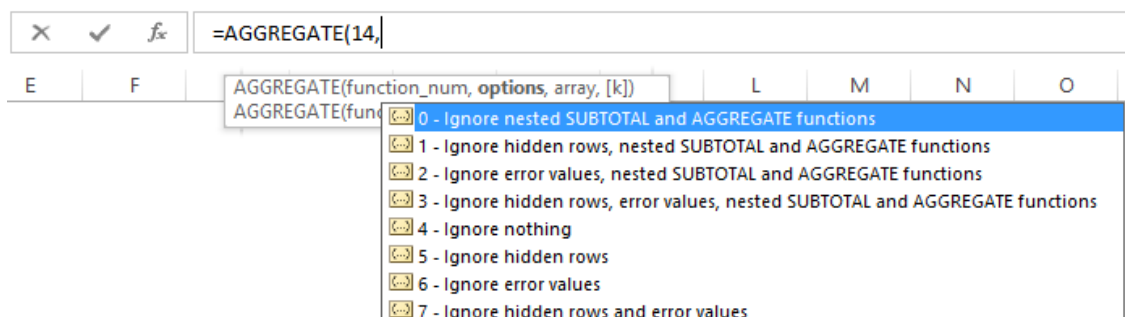
	A
1	List
2	5
3	-2
4	12
5	#REF!
6	dog
7	#DIV/0!
8	#NAME?
9	4
10	11
11	

From inspection, the third largest value is the amount in cell **A2** (the value '5'), but if you use the usual formula for this **=LARGE(A2:A10,3)** you will get the value **#REF!** as this is the first error that Excel comes across as it works down the list.

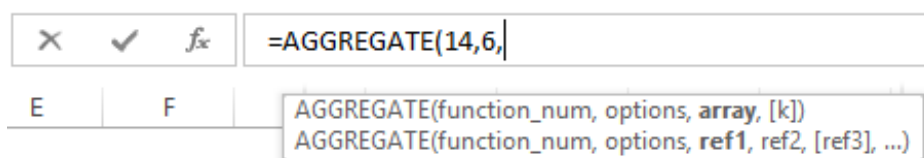
This is where you can use **AGGREGATE()** to ignore these errors. If you type in **=AGGREGATE(** you will get the following screen tip scroll list:



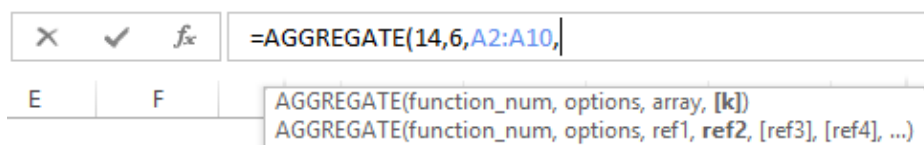
By typing '14' or selecting '14 – LARGE' from the pop-up list, you now know you are on the right track. After typing a 'comma', Excel then continues to help you:



Again by either typing a number or pointing and clicking an appropriate choice may be made. I want to ignore errors, so I need to choose '2', '3', '6' or '7', depending upon what else should be ignored. I will choose '6' – ignore error values only and then type another comma so that the screen tips keep coming thick and fast:



Now, Excel is seeking the references for evaluation. It appears to be possible that this can be in the form of a list (the array) or else discrete cell references and / or values. In this example, I will enter the range and type another comma:



Now, Excel appears to be looking for the other argument for **LARGE()** or else another reference. This is not correct. The screen tip does not update automatically. The syntax required is now just as it would if we had typed in the underlying function, *i.e.* `=LARGE(array, k)`. In this instance, this syntax always requires the fourth value to be **k**, the integer denoting the **k**th largest item in the list.

In this example, I will just type the value '3' and close brackets. Therefore, we arrive at the following formula,

=AGGREGATE(14, 6, A2:A10, 3)

which generates the correct answer '5'. The formula might look counterintuitive, but Excel has helped us every step of the way. As my oft-misquoted English teacher always used to say, practice makes perfect.

Word to the wise

Like **SUBTOTAL**, the **AGGREGATE** function is designed for columns of data (vertical ranges), not for rows of data (horizontal ranges). For example, when you subtotal a horizontal range using option one [1], such as `AGGREGATE(1, 1, ref1)`, hiding a column does not affect the aggregate sum value, although hiding a row in vertical range does affect the aggregate.

If a second **ref** argument is required but not provided, **AGGREGATE** returns an **#VALUE!** error. If one or more of the references are three-dimensional references, **AGGREGATE** returns the **#VALUE!** error value.

Beat the Boredom Challenge

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

*keep our readers engaged, we will continue to reproduce some of our popular **Final Friday Fix** challenges from yesteryear in this and upcoming newsletters. One suggested solution may be found later in this newsletter. Here's this month's...*

This month's challenge involves automating the generation of multiple worksheets from a data list. For example, we have a data list that contains the names of the worksheets that we are going to create, as shown below:

Worksheets to be created
Jul 2019 Inventory
Aug 2019 Inventory
Sep 2019 Inventory
Oct 2019 Inventory
Nov 2019 Inventory
Dec 2019 Inventory
Jan 2020 Inventory
Feb 2020 Inventory
Mar 2020 Inventory
Apr 2020 Inventory
May 2020 Inventory
Jun 2020 Inventory

We need to create multiple worksheets with the names shown in the data list above, so that the result will look like this:



We want to automate this process without manually adding each worksheet and changing the worksheet name individually. A job for VBA, perhaps..? Sounds easy? Try it. One solution *just might* be found later in this newsletter – but no reading ahead!

Charts and Dashboards

It's time to chart our progress with an introductory series into the world of creating charts and dashboards in Excel. This month, we discuss dashboard design [nice alliteration – Ed.].

The key to a good dashboard / report is creating one that is tailored to the end users, so that it fulfils all their needs. For example, there is no point for a sales team to be bombarded with other information that they do not need to do their job, such as:

- depreciation analysis
- debt / equity ratios
- budget variances in costs
- segmental margins analysis,

if it isn't relevant.

The same point goes for all other staff, so that they have more time to focus on doing their job, be it to order more stock or provide analytical reports. That's not to say this data is not essential to management, but a key idea in developing your narrative is determining your audience.

Therefore, we need to be able to identify who they are and what we think they will need to see. When identifying our target audience, we should consider the following:

- **Granularity:** how detailed the data has to be for them to do their job
- **Level of information required:** what sort of information do they need to accomplish their goal? (high level overview or more detailed?)
- **Timescale:** how much data do they need to achieve their task
- **Leading and lagging metrics:** are historical metrics sufficient or are projections required as well?
- **Decision making required:** which decisions they must make after looking at the data.

For instance, sales teams and lower management probably need to see day to day inventory stock and day to day customer traffic amounts, *e.g.*

- **Granularity:** high granularity of data
- **Level of information:** very low
- **Timescale:** short
- **Leading and lagging metrics:** historical data and short-term projections
- **Decision making required:** stock ordering, economic order quantities.

Meanwhile, manager level staff will probably need to see monthly, or even quarterly, sales and inventory stock amounts for their reports:

- **Granularity:** medium
- **Level of information:** overview of sales performance
- **Timescale:** medium
- **Leading and lagging metrics:** this is more likely to be summary (lagging) data, concentrating on historical results
- **Decision making required:** resource management and reporting issues to higher management.

Senior level management will want to see quarterly, or even yearly data:

- **Granularity:** low
- **Level of information:** overview of key metrics
- **Timescale:** long
- **Leading and lagging metrics:** historical summary with possibly a high-level projection
- **Decision making required:** supply chain solutions, staff performance management, reporting to top level management, restructure of branches, project approval, areas of improvement.

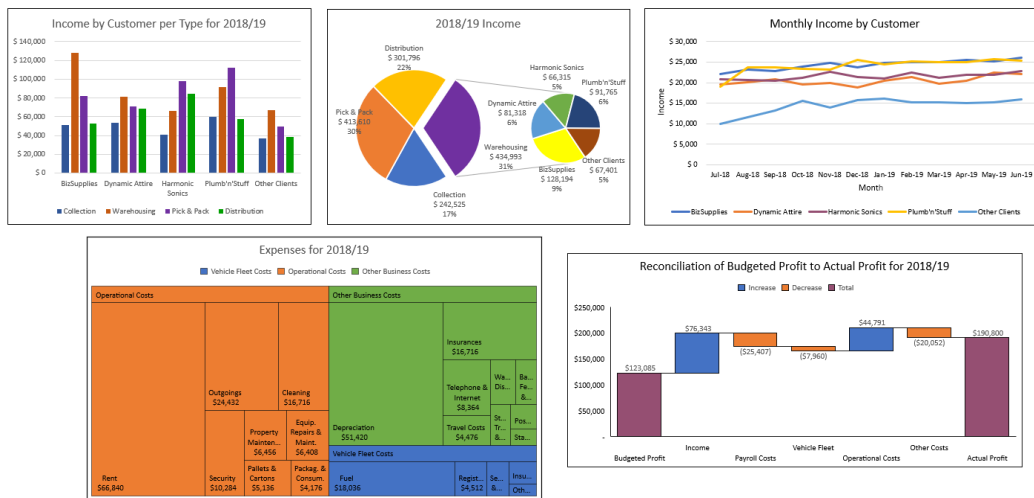
Top level management will want to see annual data and forecasts:

- **Granularity:** very low
- **Level of information:** high overview of key metrics
- **Timescale:** very long
- **Leading and lagging metrics:** historical summary, with possibly a high-level projection
- **Decision making required:** company direction, areas to grow the company, areas for company to pull out from, company investments.
- **Timescale:** very long
- **Leading and lagging metrics:** historical summary, with possibly a high-level projection
- **Decision making required:** company direction, areas to grow the company, areas for company to pull out from, company investments.

Other dashboard design considerations include which types of charts, how to use colour to your advantage and how to create memorable charts.

Now, we will consider an example of a bad and a good dashboard. For example, a manager is requesting to see metrics of their income. They also want to see some insights of the sales breakdown and performance in a financial year.

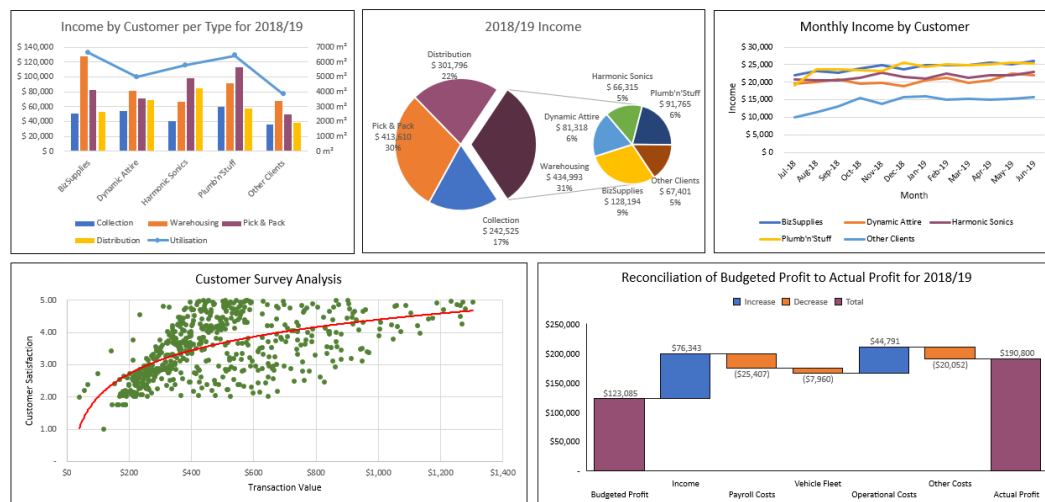
A “bad dashboard” might look like this:



Look at it. It has many colours, but it is not eye-catching:

- the charts are not organised
- the charts are not relevant to the end users
- the charts contain too much information, more than 7 + 2
- the information provided on the charts does not really help the end users
- the dashboard has arguably a terrible use of colour
- axes make results and comparisons difficult
- good point: the dashboard does actually depict seasonal trends.

Compared to the example bad dashboard, the below dashboard is enhanced:



Key things to note from this dashboard:

- a harmonious colour scheme is used
- key metrics pop out to the user: All important information can be found in five seconds
- the charts are organised
- the appropriate charts have been selected to display the respective data
- the dashboard displays seasonal trends.

More next month...

Visual Basics

We thought we'd run an elementary series going through the rudiments of Visual Basic for Applications (VBA) as a springboard for newer users. This month we go back to basics with the first in a continuing series.

VBA (Visual Basic for Applications) is a programming language developed by Microsoft and is built into Excel and other Office programs. It enables building your own functions (known as User Defined Functions or UDFs for short), automating processes and interacting with libraries to work with the Windows operating system. It is also used to write some of the custom add-ins for Office applications.

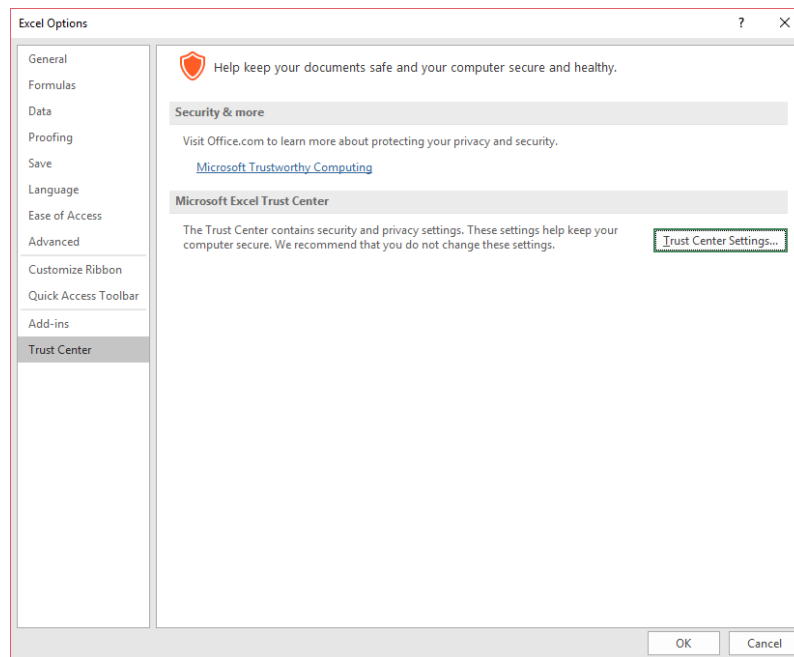
In Office for Mac, VBA was upgraded in 2016 but some features and certain commands do not function correctly and herein lies the problem: it doesn't work *everywhere* – but lots of people use it as a readily accessible porthole for Microsoft Office and often Excel in particular.

You can't say "VBA" without talking about macros. A macro is simply a sequence of actions that can be executed together, written in VBA. Office applications (apart from PowerPoint) allow the "recording of

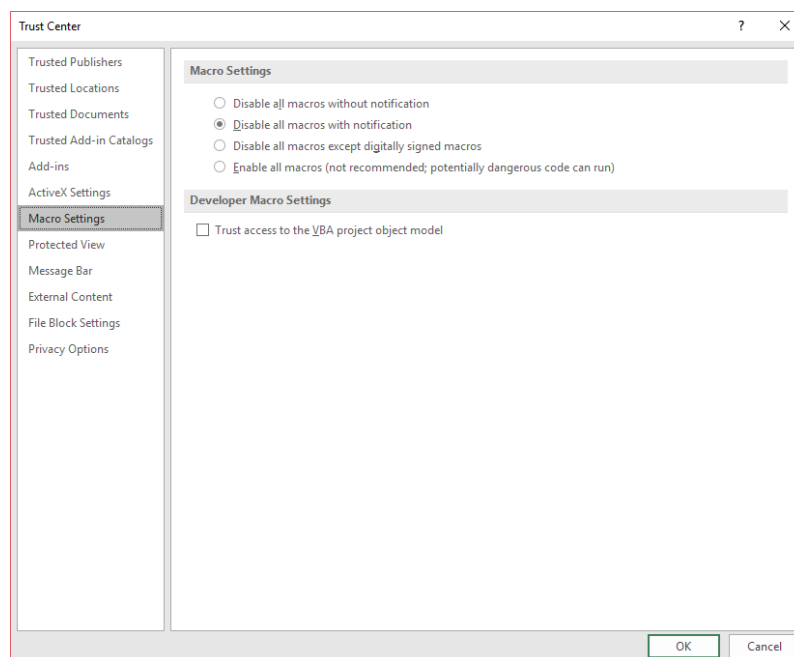
macros" where every single step performed is recorded and translated to Visual Basic code. After recording / writing a macro, these bits of code can be reused to automate these actions.

This is all well and good, but macros are disabled by default in the Office applications. This is because VBA scripts may be automatically executed whenever a spreadsheet is opened – and that's not good given all those nasty people out there. Some scripts can be malicious and cause issues on your machine. For example, some VBA scripts can delete files and your Outlook email address book.

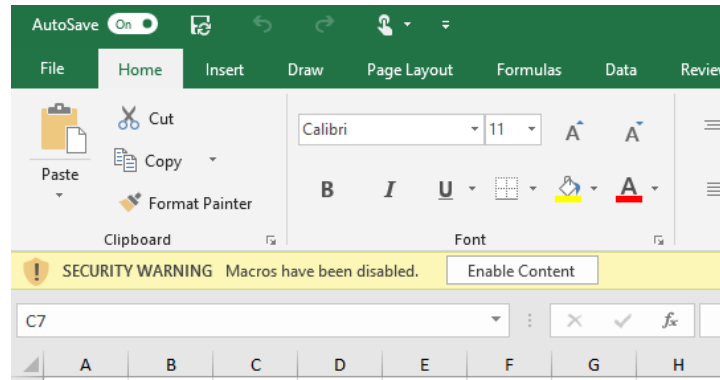
To change the security settings for macros, these may be accessed in 'Excel Options' (**ALT + T + O**) in the 'Trust Center' group in the resulting left-hand column of the dialog box. Once there, click on the 'Trust Center Settings...' button, viz.



Under the 'Macro Settings' group, a set of options is presented:

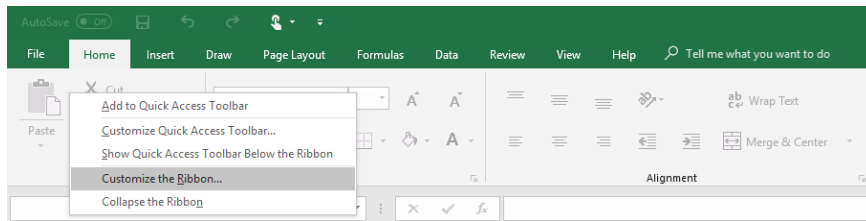


Excel defaults to 'Disable all macros with notification'. This means that whenever a workbook has a macro (only .xls, .xlsx and .xlsm files may have macros) a pop-up warning window will appear before opening a file:

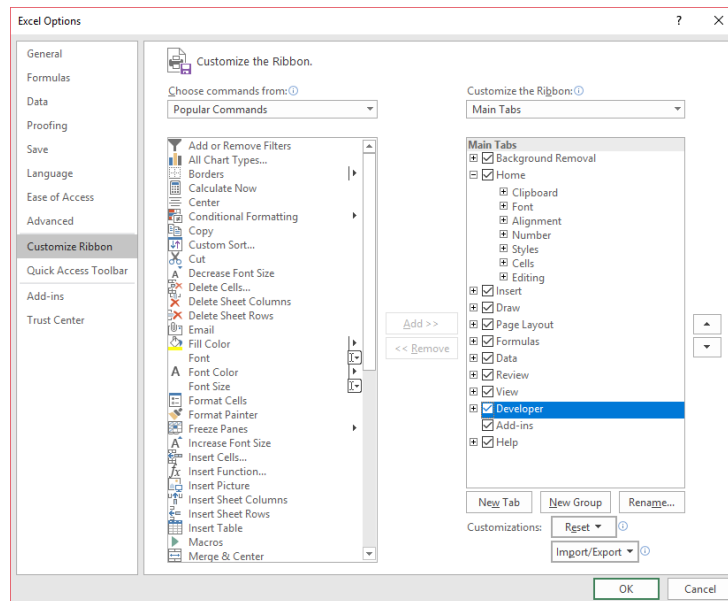


This is a little confusing for older users of Excel as the default is similar to Excel 2003, even if the setting starts with 'Disable'. It's not the only thing you have to do, mind you.

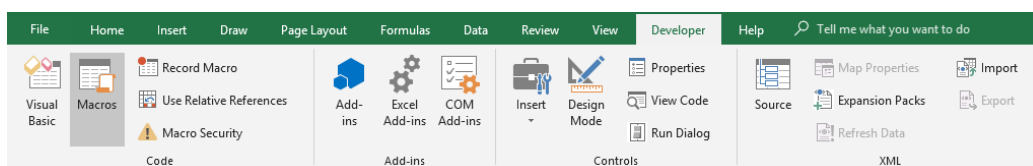
To fully utilise VBA in Excel, you need to have access to the 'Developer' tab, which shows the VBA tools available. By default, the Developer tab of the Ribbon is not made visible. To enable this as well, right-click on the Ribbon and select 'Customize the Ribbon...':



The 'Excel Options' menu will pop up. On the right-hand side, you'll see "Customize the Ribbon" where you can click which tabs are to be visible. Click on the check box next to 'Developer' and select 'OK'.



Then we'll see the 'Developer' tab on the screen.



You're good to go!

More next month.

Power Pivot Principles

We continue our series on the Excel COM add-in, Power Pivot. This month, we consider how to include filters in the MAXX function.

Last newsletter we covered how the MAXX functions works. This time, we'd like to expand on the MAXX function a little and show you how we may use other functions that return with a table to alter the MAXX function's results.

For this example, let's look at the following Table (the capitalisation is deliberate):

	A	B	C	D	E
3					
4		Customer	Customer Type	Price	Purchase Volume
5		Abiel	1	\$ 31.00	4
6		Beverly	1	\$ 10.00	10
7		Craig	3	\$ 9.00	6
8		David	1	\$ 96.00	5
9		Frank	2	\$ 69.00	10
10		Ellie	2	\$ 85.00	4
11		Niel	2	\$ 77.00	3
12		Maddie	1	\$ 18.00	10
13		Henry	2	\$ 91.00	7
14		Lester	2	\$ 5.00	5
15					

As a quick refresher, the following measure calculates the greatest amount spent on apples:

```
=MAXX(  
    'AppleSalesCustType',  
    'AppleSalesCustType'[Price] * AppleSalesCustType[Purchase Volume]  
)
```

Measure

Table name: AppleSalesCustType

Measure name: MAXX

Description:

Formula: `=MAXX('AppleSalesCustType', 'AppleSalesCustType'[Price]*AppleSalesCustType[Purchase Volume])`

Formatting Options

Category: Number

Format: Decimal Number

Decimal places: 2

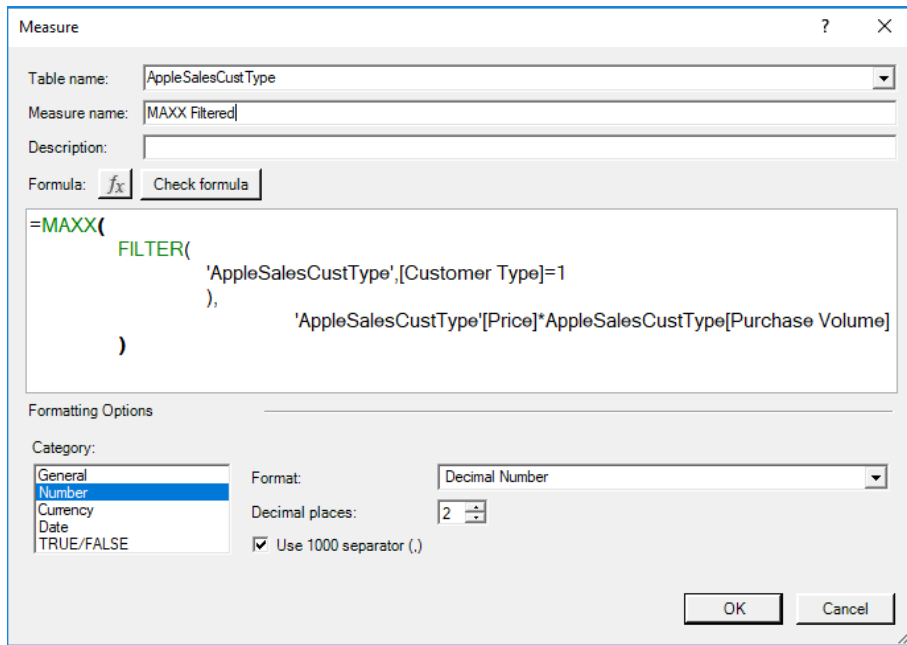
Use 1000 separator (,)

OK Cancel

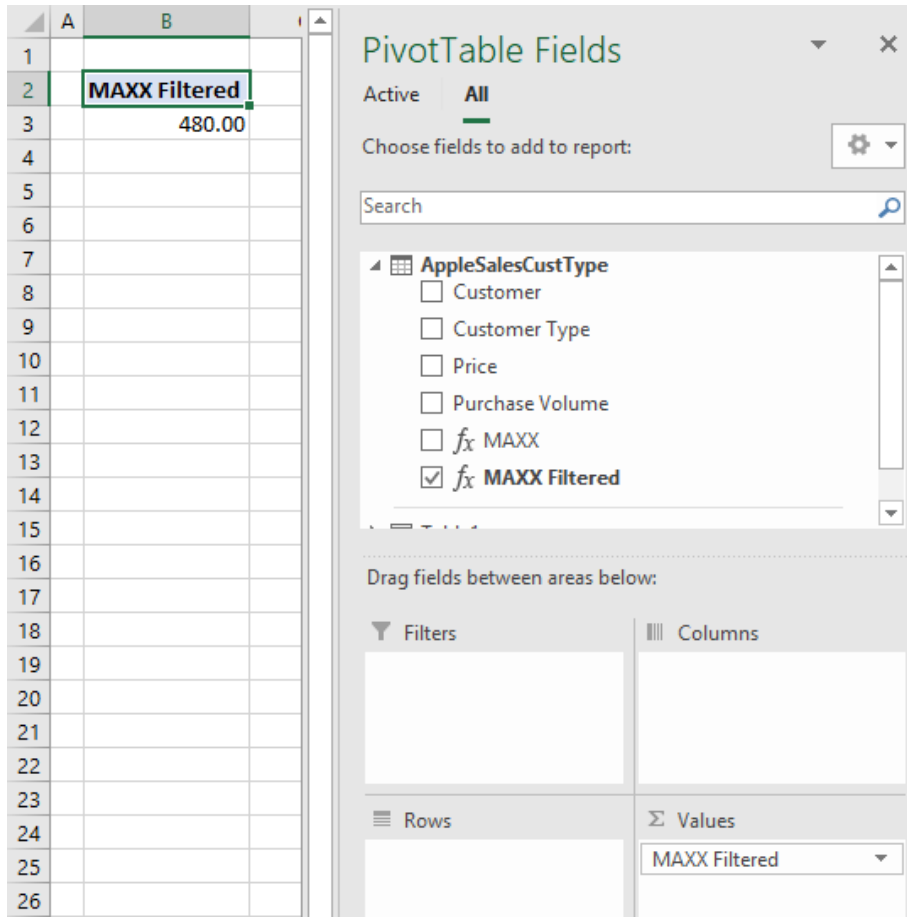
This measure returns with the value of \$690.00, which is the greatest amount spent on apples without any filters.

What we did not cover last time is that the MAXX function can work with other functions too. For instance, we may use the FILTER function to filter out customers that do not fulfil the criterion 'Customer Type' = 1.

```
=MAXX(  
    FILTER(  
        'AppleSalesCustType',[Customer Type]=1  
    ),  
    'AppleSalesCustType'[Price] * AppleSalesCustType[Purchase Volume]  
)
```



The **FILTER** function returns with a **table** (quoting the **MAXX** syntax from last time). Therefore, we are able to use these two functions together. Exporting this to a PivotTable yields the following result:



The greatest amount spent on apples by customers who fall into the '1' category is \$480.00.

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

Power Query Pointers

Each month we'll reproduce one of our articles on Power Query (Excel 2010 and 2013) / Get & Transform (Office 365, Excel 2016 and 2019) from www.sumproduct.com/blog. If you wish to read more in the meantime, simply check out our Blog section each Wednesday. This month, we look at some useful logical **M** functions which convert data into dates or vice versa.

To conclude our series on dates, let's take a look at some functions that we may use to convert a date into a different data type. For example, one such data type would be text, so that we may use a date as part of a reference. We will also look at functions that perform the opposite process, converting dates into text.

Date.FromText

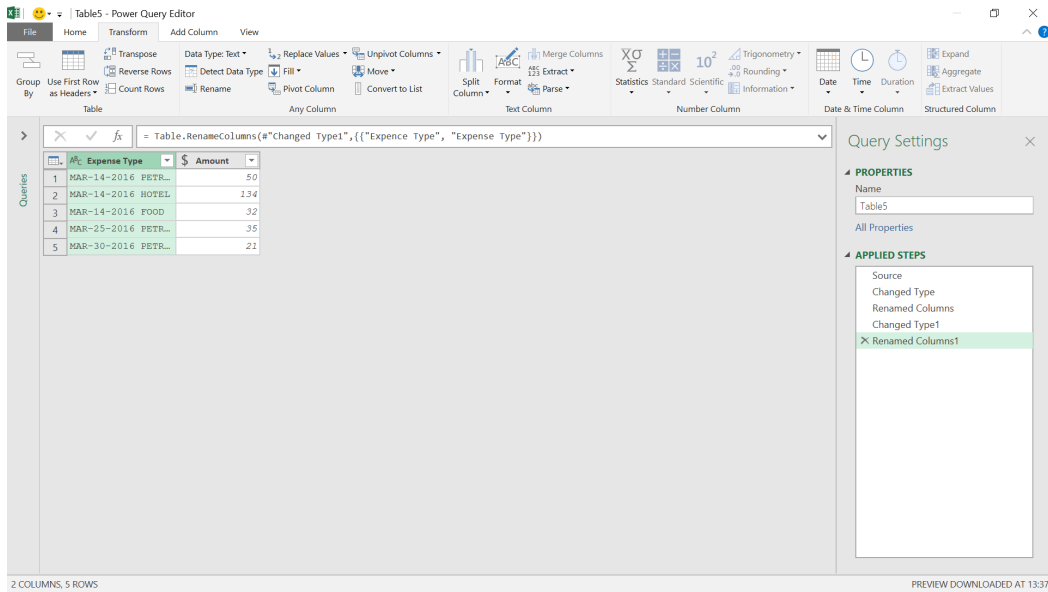
Date.FromText(date as nullable text, optional culture as nullable text) as nullable date

This returns a date value from text (**date**) in a recognised date format according to the **culture**, following the ISO 8601 format standard.

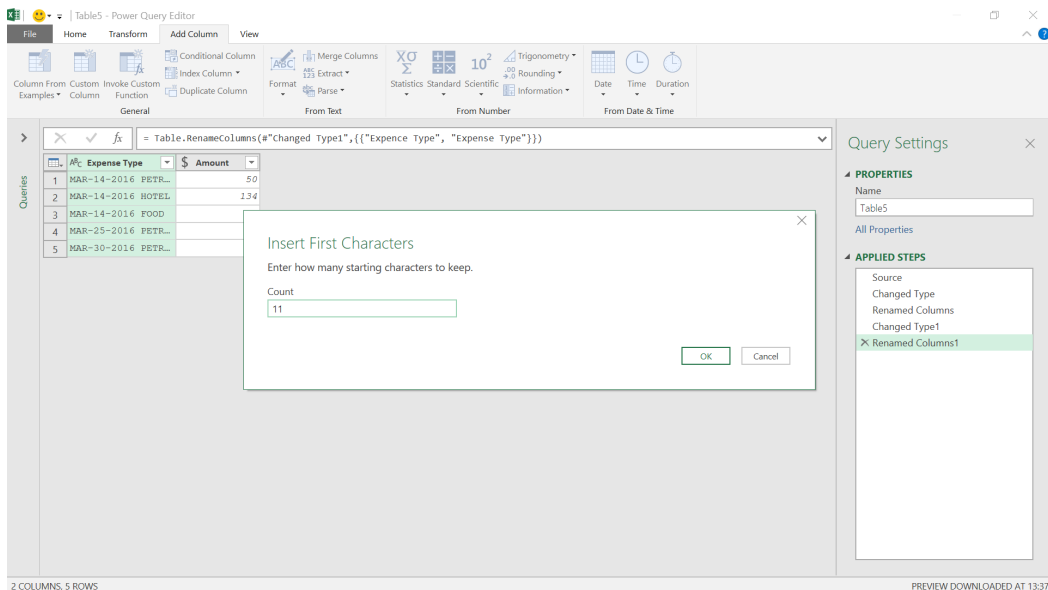
It's probably helpful to know what formats would be accepted here. For en-US culture, text in the form **yyyy-MM-dd** is accepted, as is **yyyyMMdd** and **M/d/yyyy**. Cultures such as the UK and Australia prefer

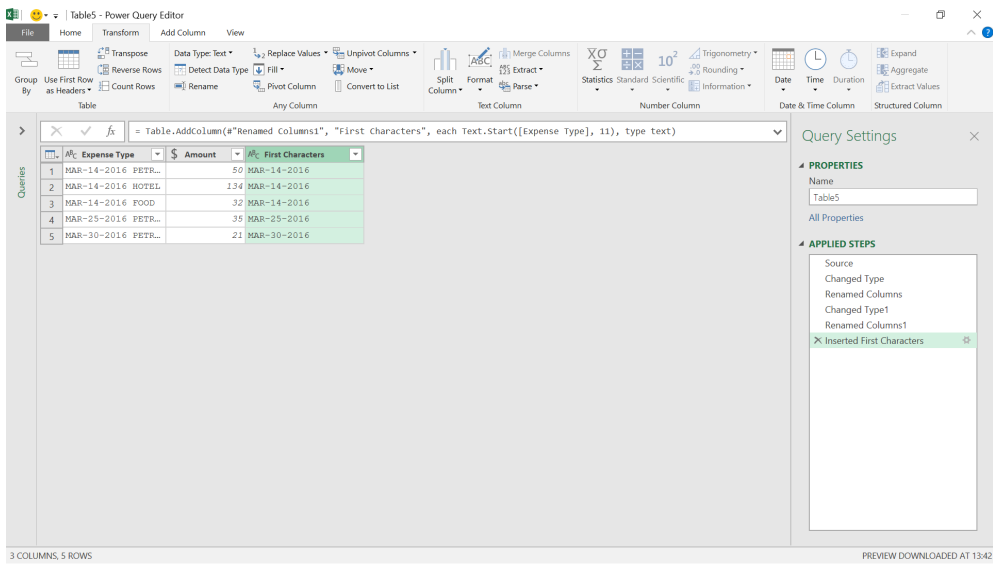
to put the day before the month, so it helps to know the windows culture being used: this may be specified in any of the formulas we are looking at today.

As an illustration, we have some data from our fictional sales person, John. He has decided to embed the date in a column and we need to extract it and convert it into data type date so that we may link the data to other tables.

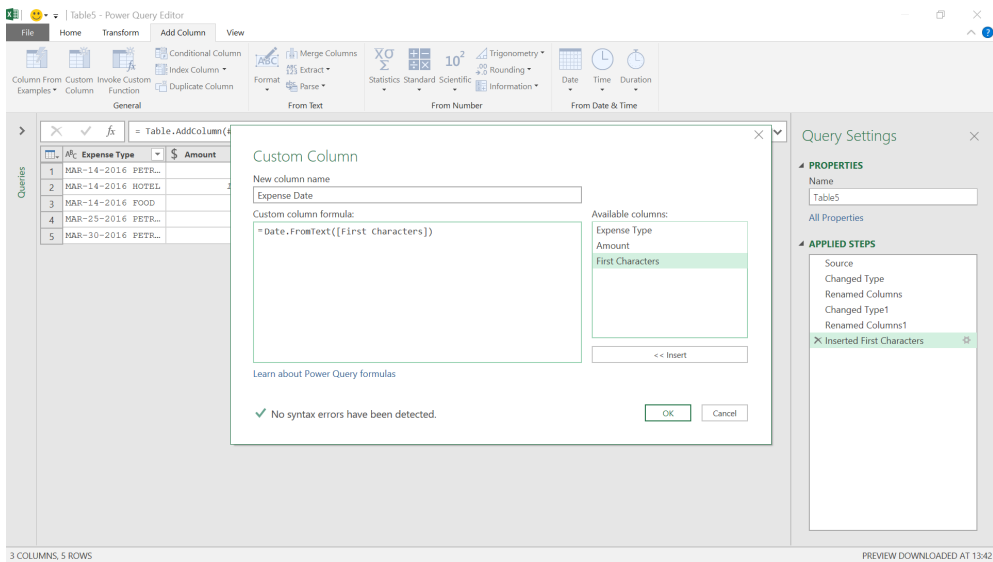


The current **Expense Type** is a text field that contains the date. We want to extract the date part of the text column and convert it to a date. We'll begin by extracting the date using the 'Extract' option in the 'From Text' section on the 'Add Column' tab.



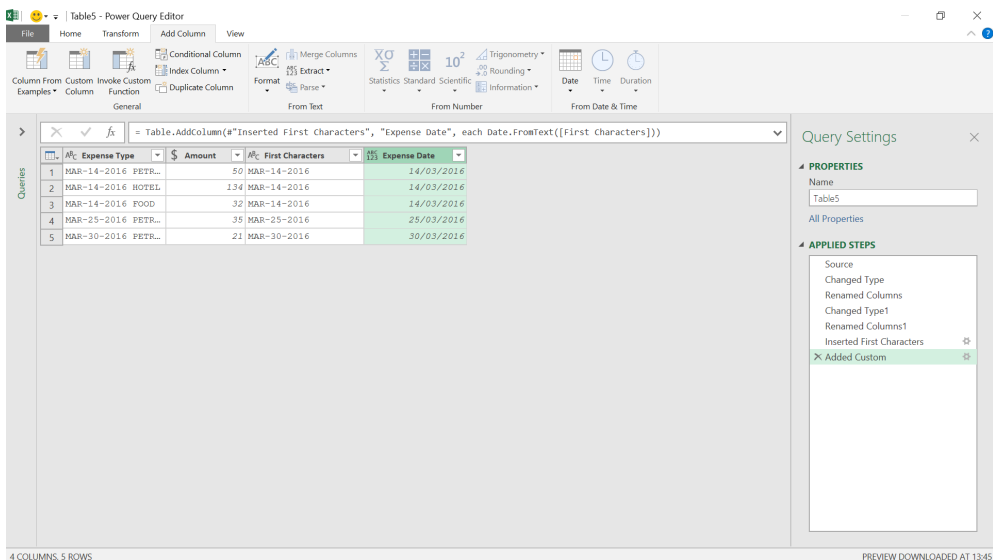


We could now just convert this to a date by using the Transform menu, but instead we will enter the formula directly to create a new custom column.

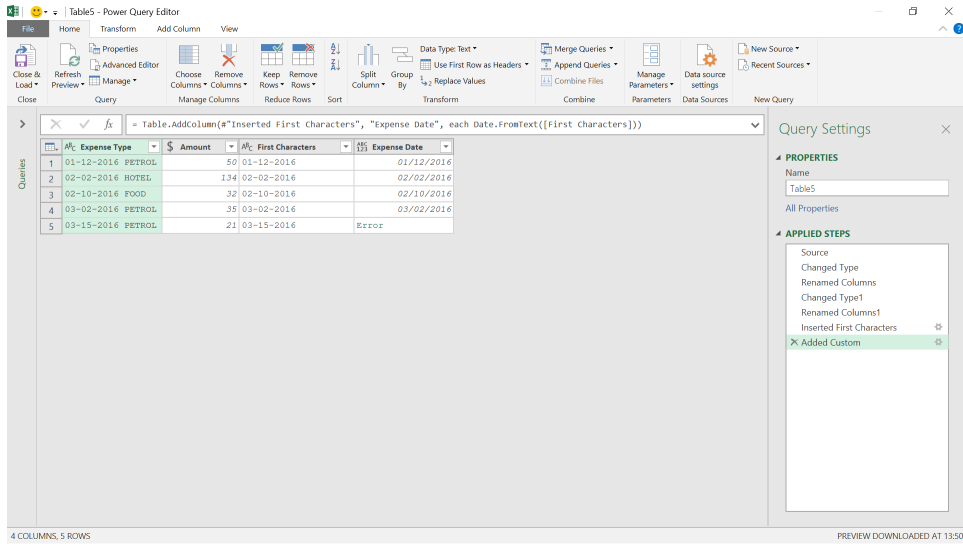


The M formula used is

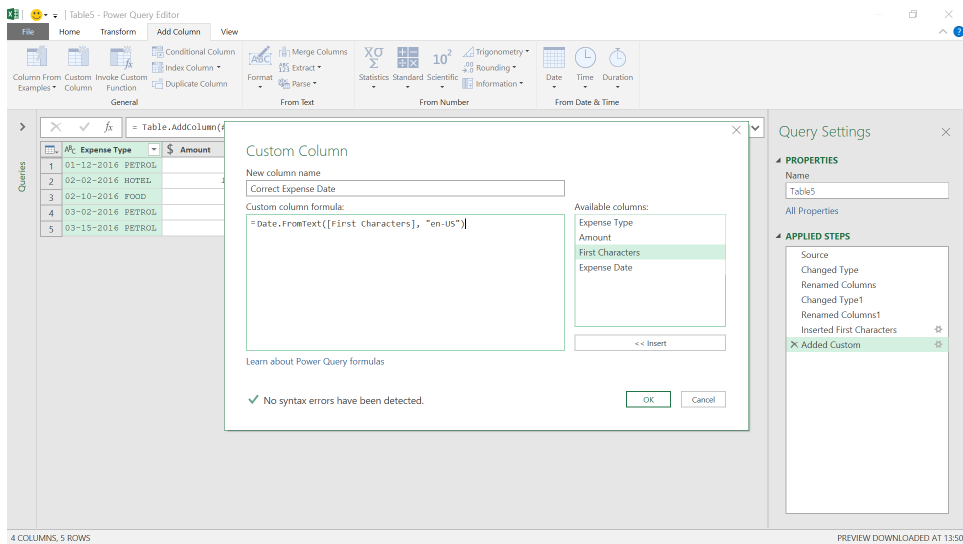
= Date.FromText([First Characters])



The date has been converted correctly. But why would we ever need to use a formula instead of just using a transformation? Suppose we had the next dataset coming in...

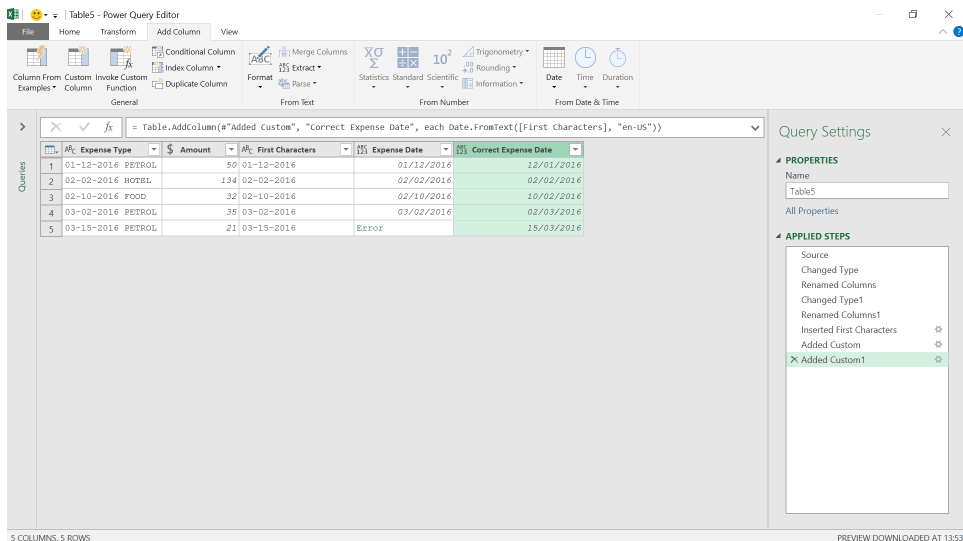


Well that doesn't look good. We need to alter the **Expense Date** formula. We will add a new column that will tell Power Query to use en-US culture, as John has clearly used US date formatting.



The formula used is:

= Date.FromText({First Characters}, "en-US")



The date has been calculated correctly because we have specified the culture used.

Date.ToText

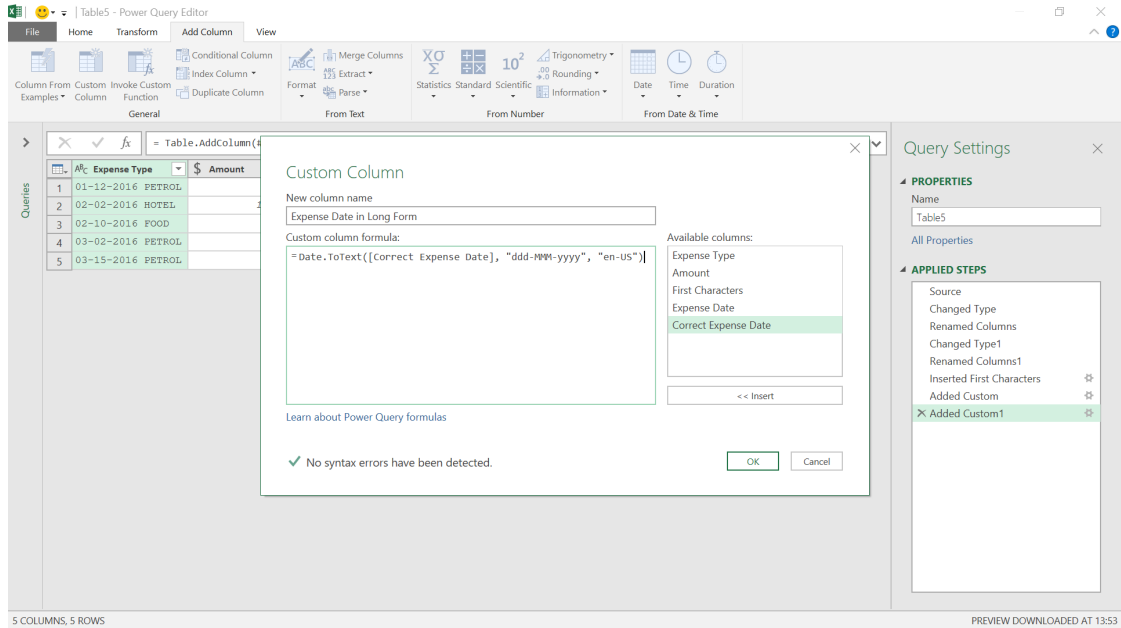
Date.ToText(date as nullable date, optional **format** as nullable text, optional **culture** as nullable text) as nullable text

This returns a textual representation of **date**. This function takes in an optional **format** parameter.

The full list of formats that are supported are supposedly in the Microsoft

help pages, but Lord Lucan is a little easier to find; we'll give a couple of examples in the exercise that follows.

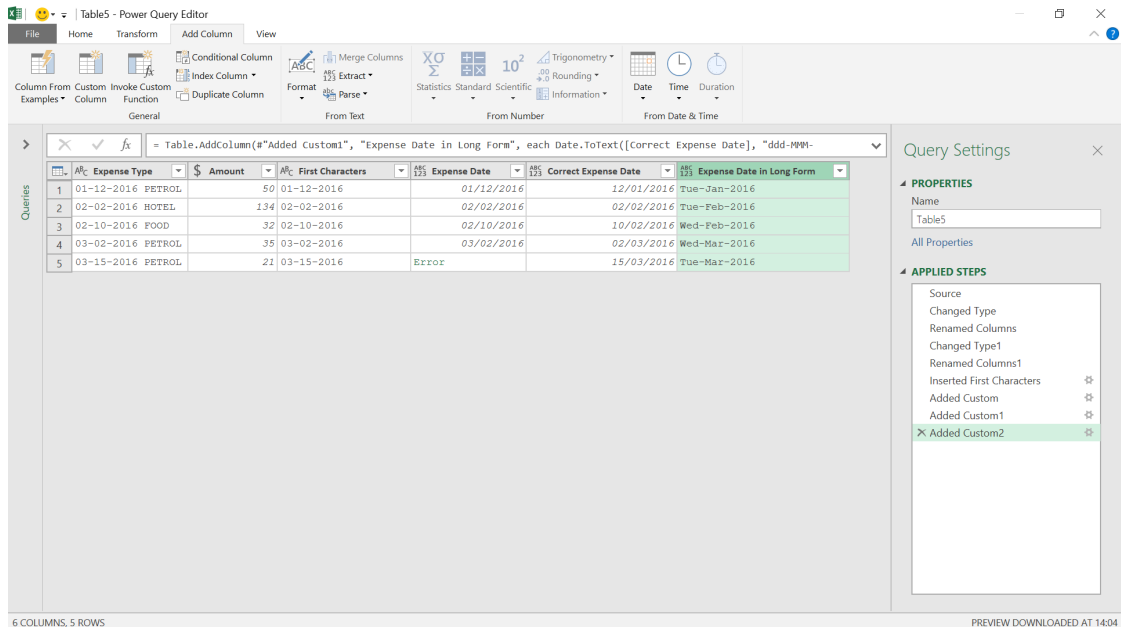
We may use this **M** function to format dates ready to add them to a reference or to display them in a required format for a report. We will format **Correct Expense Date** in a couple of ways to show how this function can be used.



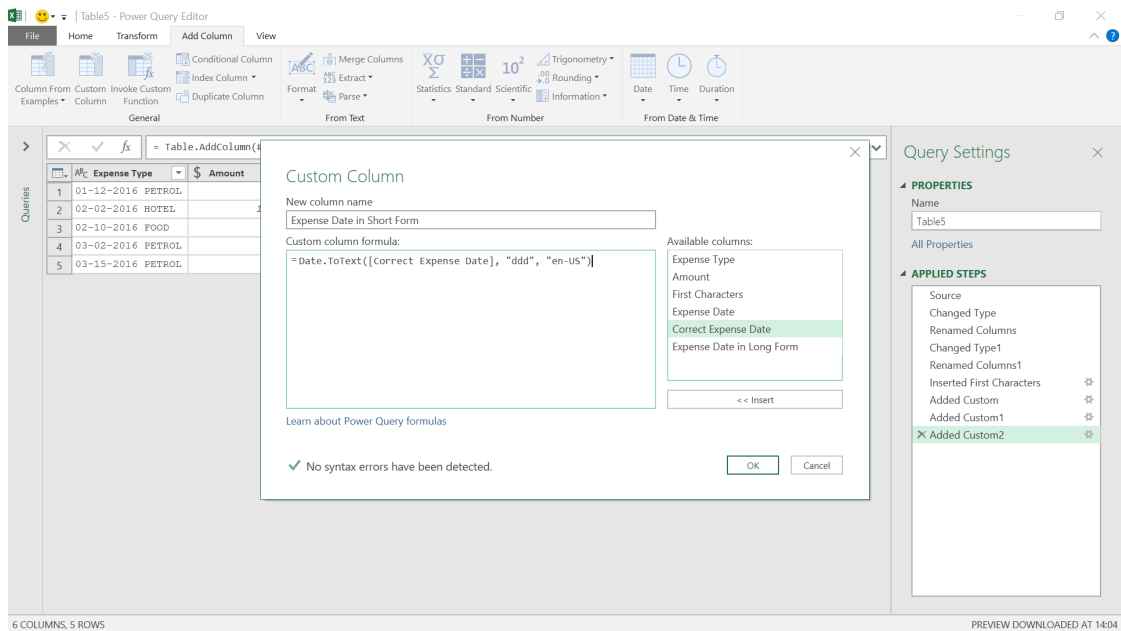
The **M** formula used is

= Date.ToText([Correct Expense Date], ddd-MMM-yyyy, "en-US")

We have to mention the **culture** because the date is held using the US date format.

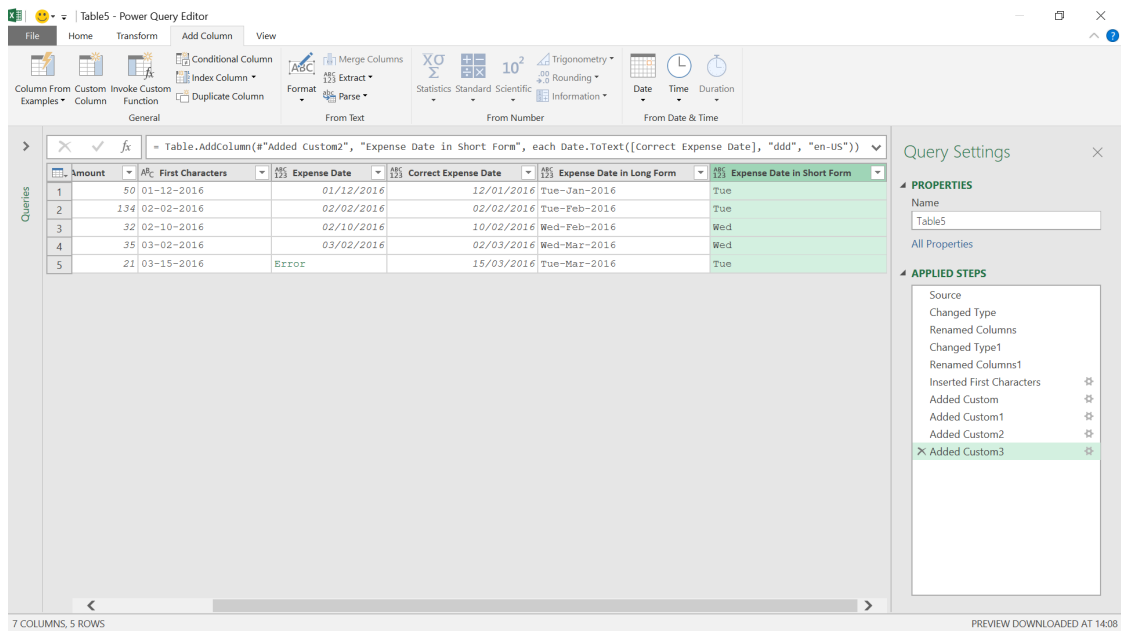


Now we will use a shorter form.



The M formula used is

= Date.ToText([Correct Expense Date], "ddd", "en-US")



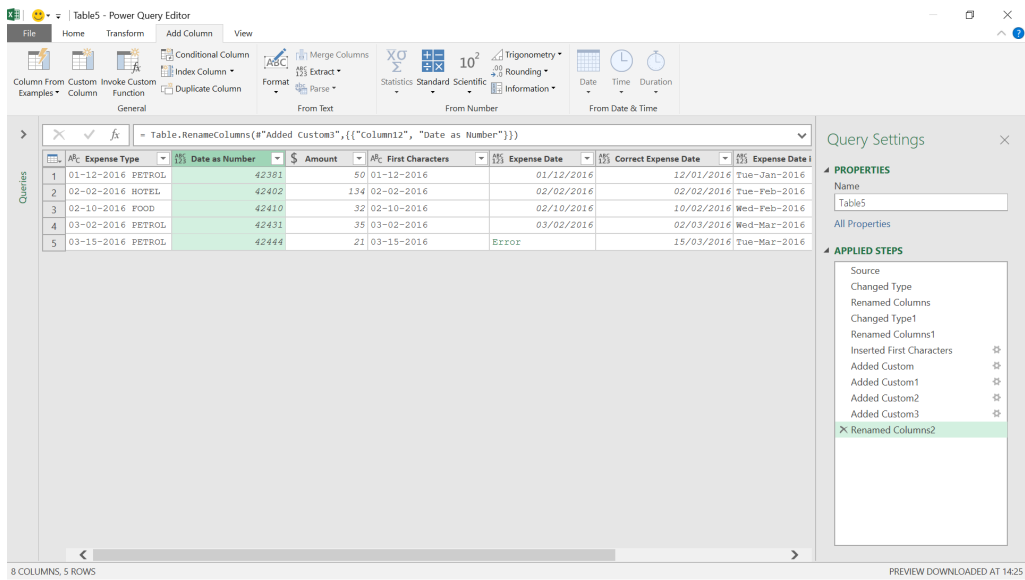
This gives an idea of the range of different texts that may be produced.

Date.From

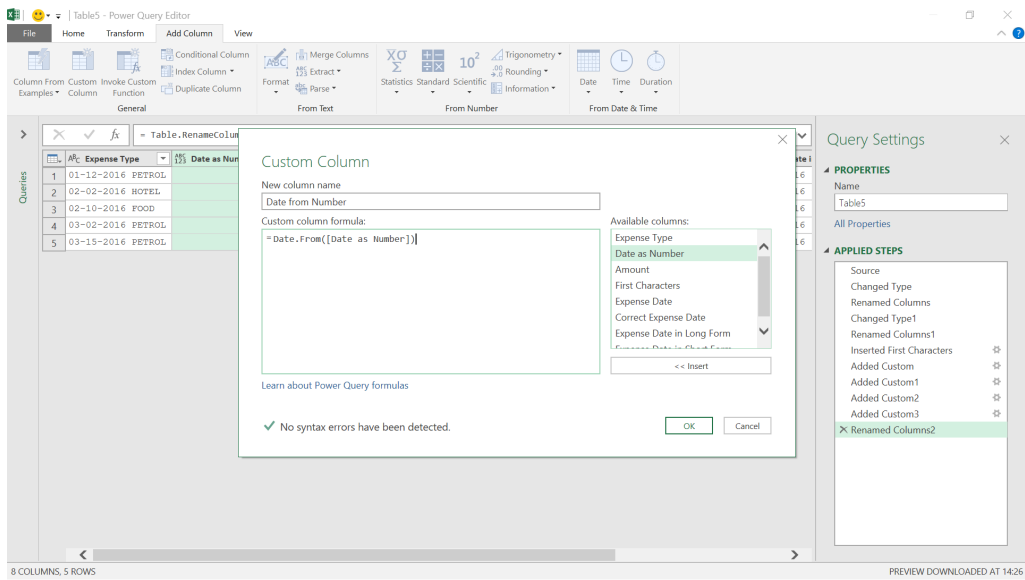
Date.From(value as any, optional culture as nullable text) as nullable date

This returns a date value from a **value**.

Similar to **Date.FromText()**, the main advantage that this function has over the standard menu option of transforming a number to a date, is that it can specify the **culture**. Below, a column has been inserted that contains the expense date in a numeric format:

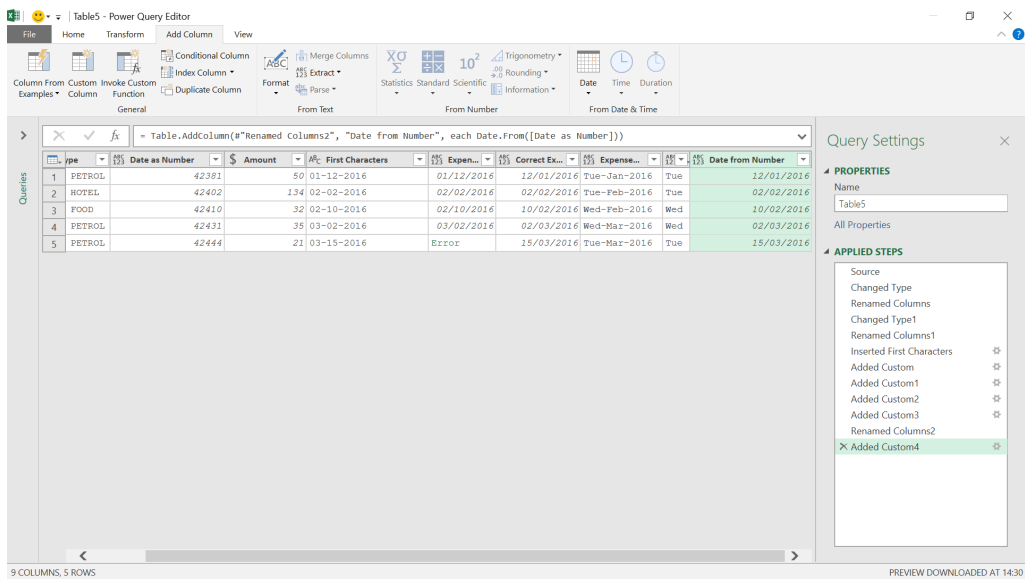


We add a new column where the number is converted to a date.



The M formula used is

=Date.From([Date as Number])



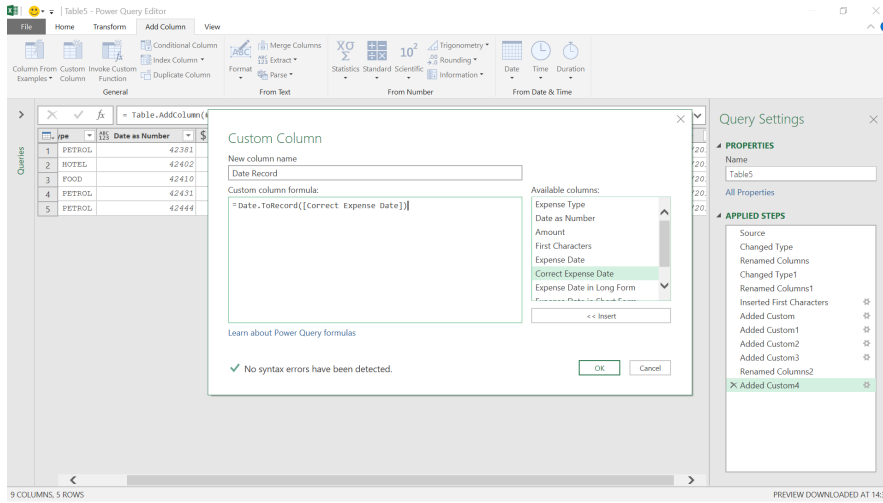
The date is shown in its proper format for our system.

Date.ToRecord

Date.ToRecord(date as date) as record

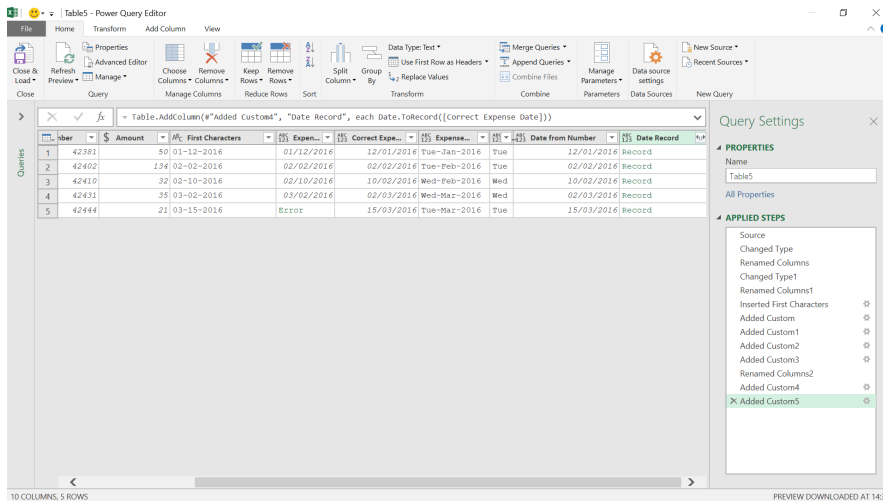
This returns a record containing parts of a date value.

This is more for the techies amongst us, as it will create a record containing the year, month and day as entries. We will do this for **Correct Expense Date**.

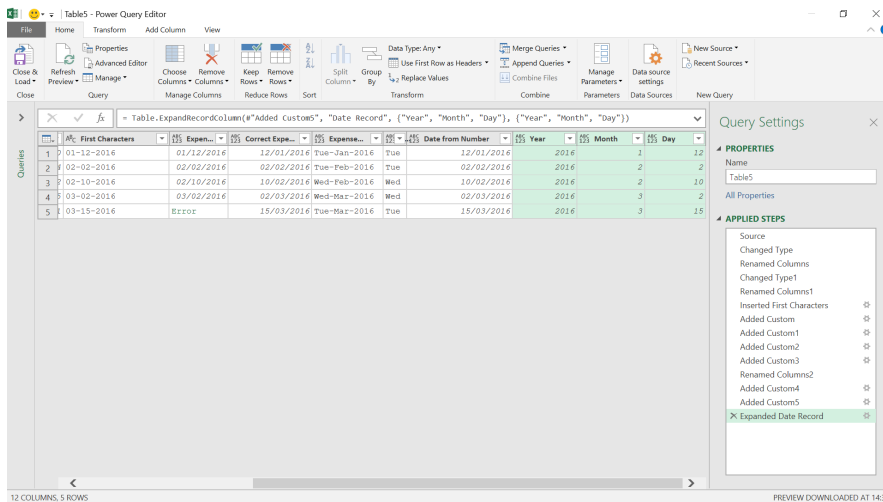


The M formula used is

= Date.ToRecord([Correct Expense Date])



We can see the records and what is in them by expanding the **Data Record** column:



This is a nice way to extract the year, month and day all in one move.

More next time.

Power BI Updates

It seems a little quieter this month – not surprising given how much we reported in last month’s newsletter! Nevertheless, let’s get you up to date. The full list is as follows:

Reporting

- Smoothed and Leader lines – updates
- On-Object Interaction – updates in Preview:
 - Customise the Pane Switcher
 - **CTRL Click** to open multiple panes in the Pane Switcher
 - Treemap sub-selections now supported
 - Noteworthy bug fixes

Modelling

- Edit your Data Model into the Power BI Service – updates in Preview:
 - Relationship validation

Data connectivity

- Google Analytics (connector update)
- Oracle Database (connector update)
- Azure Databricks / Databricks (connector update)
- Denodo (connector update)
- EQuIS (connector update)
- Anaplan (connector update)

Service

- Dataset details page revamp

Mobile

- Datasets are coming to Power BI Mobile Apps

Developers

- Boost your Custom Visuals performance

Visualisations

- New visuals in AppSource
- Elevate Your Map Charts with Drill Down Map PRO
- “All-in-one” Multi Target KPI card

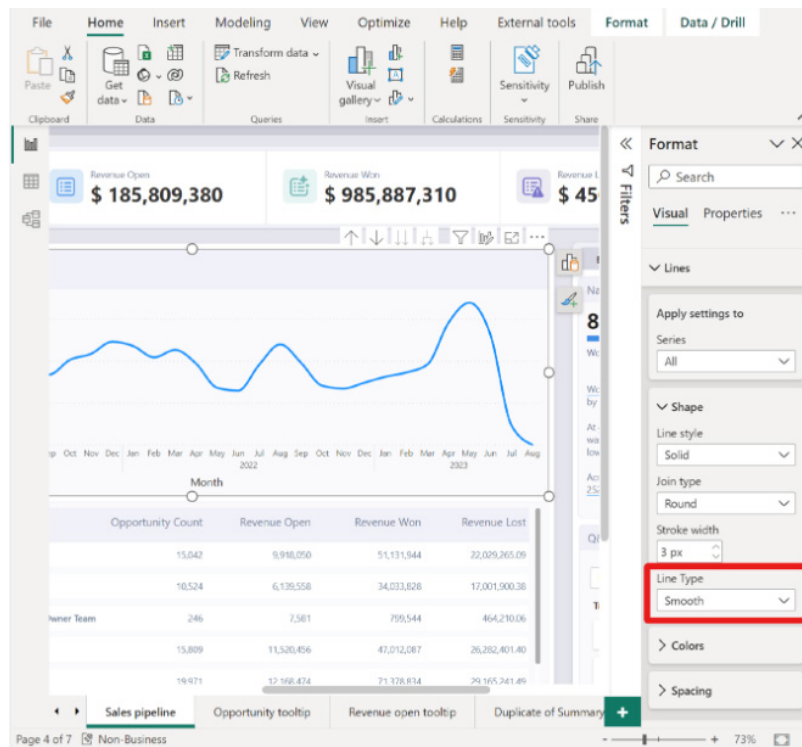
Other

- WebView2 now Generally Available.

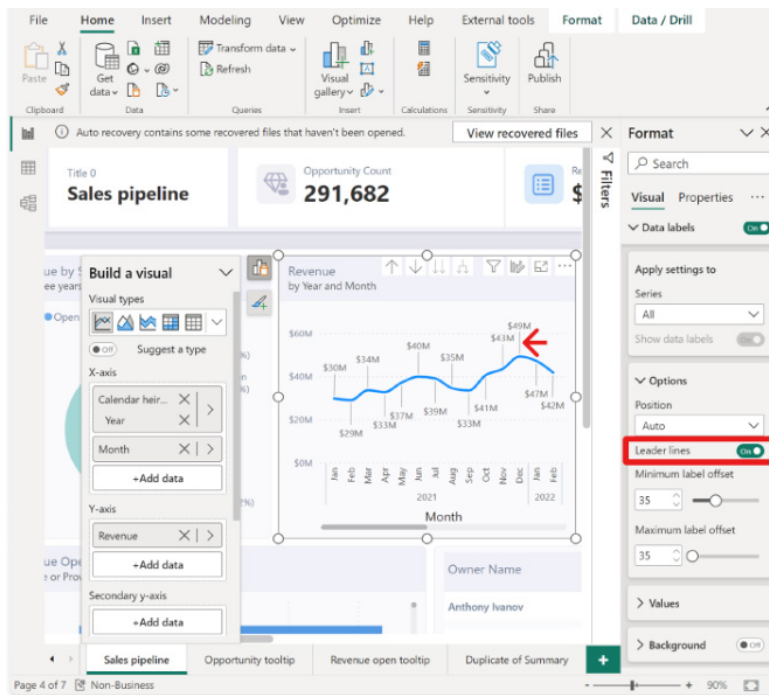
Let’s look at each in turn.

Smoothed and Leader lines – updates

Report creators may now create smoother line and area charts, providing a more polished look to their visualisations. To access this setting, go to **Lines -> Shape -> Line Type**:



Power BI has recently added leader lines for both line and area charts. This new feature creates a visual connection between each data point and its corresponding label. To access this feature, simply navigate to the **Data labels -> Options -> Leader lines**.



Apparently, these features are just the beginning of several improvements coming soon for graphs, charts, plots and markers.

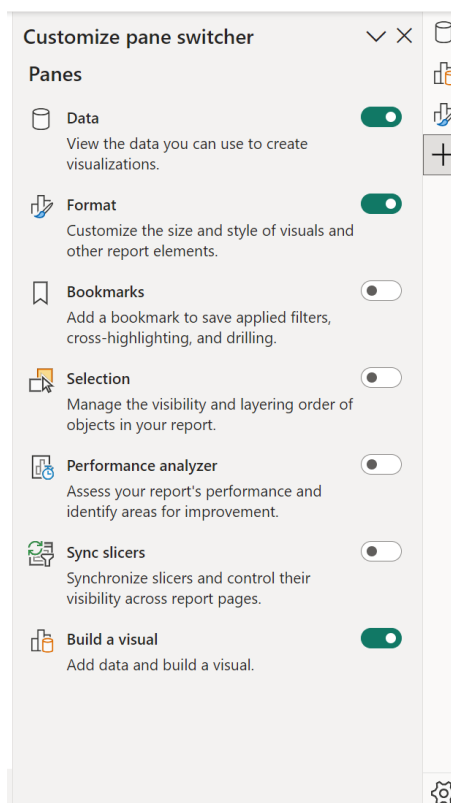
On-Object Interaction – updates in Preview

The new On-Object Interaction feature was released in Preview back in March. This month sees more improvements and bug fixes.

CUSTOMISE THE PANE SWITCHER

Power BI has now added a new '+' button on the Pane Switcher to quickly add new panes directly from the Pane Switcher without having to go to the View tab on the Ribbon. This menu also provides you with a brief

description of what panes are available and what their functions are. Furthermore, the panes added to the Pane Switcher are saved across reports. Therefore, you only need to set configurations once.



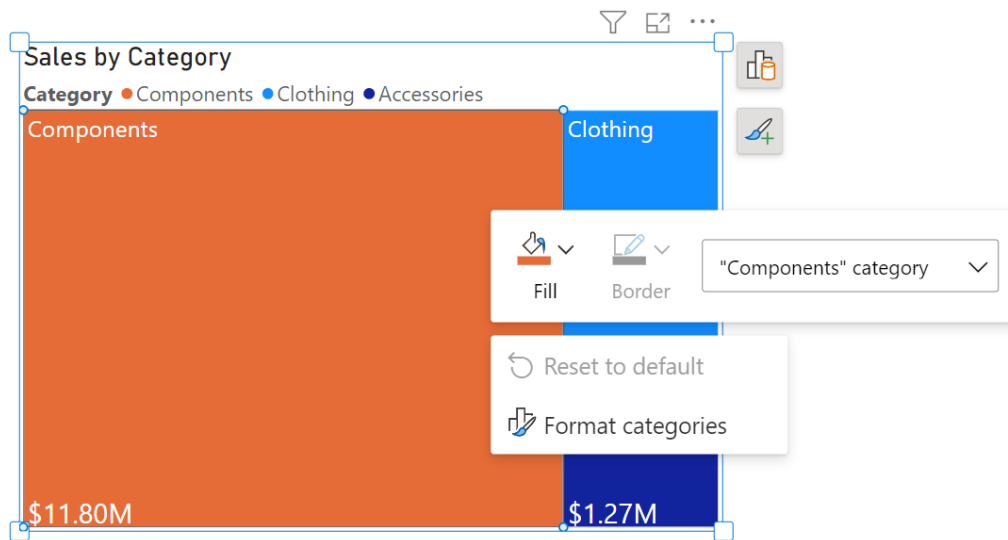
You can also access the two [2] preference settings released in last month's update to 'Always show the Pane Switcher' and re-attach the Build menu as a pane by using the gear icon.

CTRL CLICK TO OPEN MULTIPLE PANES IN THE PANE SWITCHER

In addition to the right-click option 'Open in new pane', it is now even easier to open multiple panes from the Pane Switcher by simply holding down the **CTRL** key and **Clicking** the pane you wish to open.

TREEMAP SUB-SELECTIONS NOW SUPPORTED

As demonstrated below:



NOTEWORTHY BUG FIXES

These include:

- overlap of the On-Object buttons on the Formula bar has finally been resolved
- the visual ToolTip shows when opening the Build menu automatically, blocking the Formatting On-Object button is now fixed
- selected visual type is now reflected in the Ribbon visual gallery accordingly
- the mini-toolbar's fill colour icon now reflects conditional formatting gradient as well.

Edit your Data Model into the Power BI Service – updates in Preview

Data Model editing in the Power BI Service feature was released in Preview back in April. Microsoft has made one improvement this month...

RELATIONSHIP VALIDATION

Microsoft is adding relationship validation into the Service, making it easier to create and edit relationships on the web. Like Power BI Desktop, as you define the properties of your relationship, the system will automatically validate it and offer appropriate choices for cardinality and cross filter selections.

Google Analytics (connector update)

The Google Analytics connector has been updated to support Google Analytics Data API (Google Analytics 4). To use this new functionality, use 'Implementation 2.0' when connecting. Existing connections will not be affected.

Oracle Database (connector update)

The Oracle connector has been updated to enable Azure AD-based Single Sign-On functionality through the on-premises data gateway. This will require the July release of the on-premises data gateway.

Azure Databricks / Databricks (connector update)

The Azure Databricks and Databricks connectors have been updated. The following amendments have been made:

- added a new DSRHandler to databricks-multicloud
- fixed C_NOT_ENABLED and Catalog 'spark' not found' error in legacy code path using Databricks.Contents.

Denodo (connector update)

This new version adds graphical support for the specification of native SQL queries at data source creation time.

EQulS (connector update)

The EQulS connector has been updated. The following adjustments have been made:

- removed 'Beta' attribute
- retrieved report content as .csv to remove the row limitation of .xlsx files
- optimised handling of facility groups in the Navigation tree
- now show report and / or location folders in Navigation tree, even if one or the other is empty.

Anaplan (connector update)

This version of the Power BI connector for Anaplan includes backend changes for compatibility with ongoing Anaplan infrastructure updates. However, there is no change to user facing connector features.

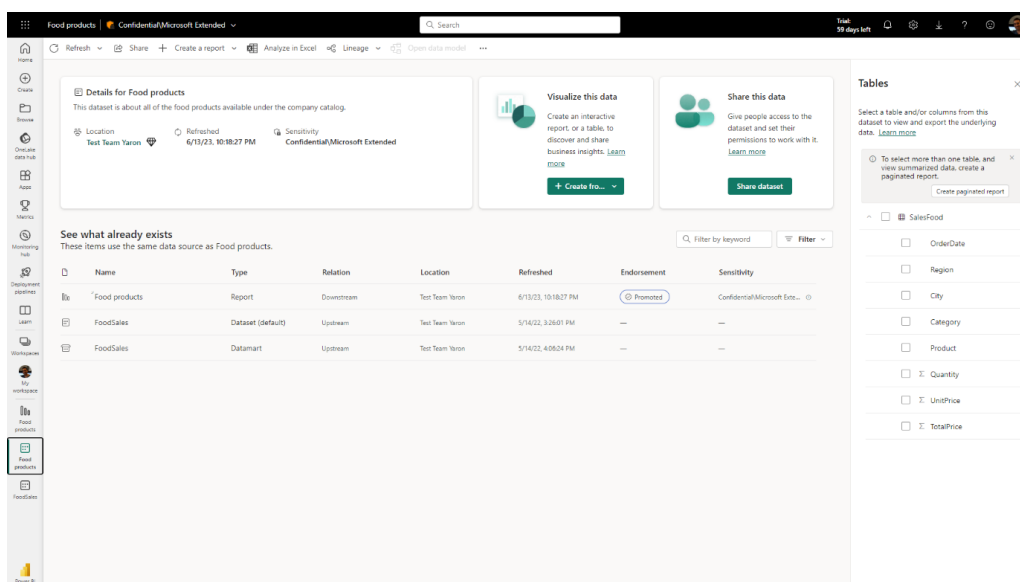
Dataset details page revamp

Now, when you click on a dataset item in the OneLake data hub and workspace view, you will be directed to the redesigned page that not only enhances the look and feel but also introduces new capabilities for an improved user experience.

Here's what you can expect to find on the dataset details page:

- **Actions:** you will find various actions that can be performed on the dataset, such as creating a report and refreshing the dataset. With this release, there is now the option to view the Refresh history under the Refresh menu
- **Dataset Metadata:** gain insights into the dataset through its description and last refresh time
- **Related Items:** explore existing related items associated with the dataset
- **Dataset Schema:** obtain a comprehensive view of the dataset's tables and columns. Clicking on a table provides a table preview, with export capabilities available using paginated reports behind the scenes.

Additionally, there are now significant improvements to the related items list. It now showcases all the downstream and upstream dependencies for the dataset. This enhancement allows you to easily identify the sources of the dataset, composite model relations, reports and dashboards associated with it.



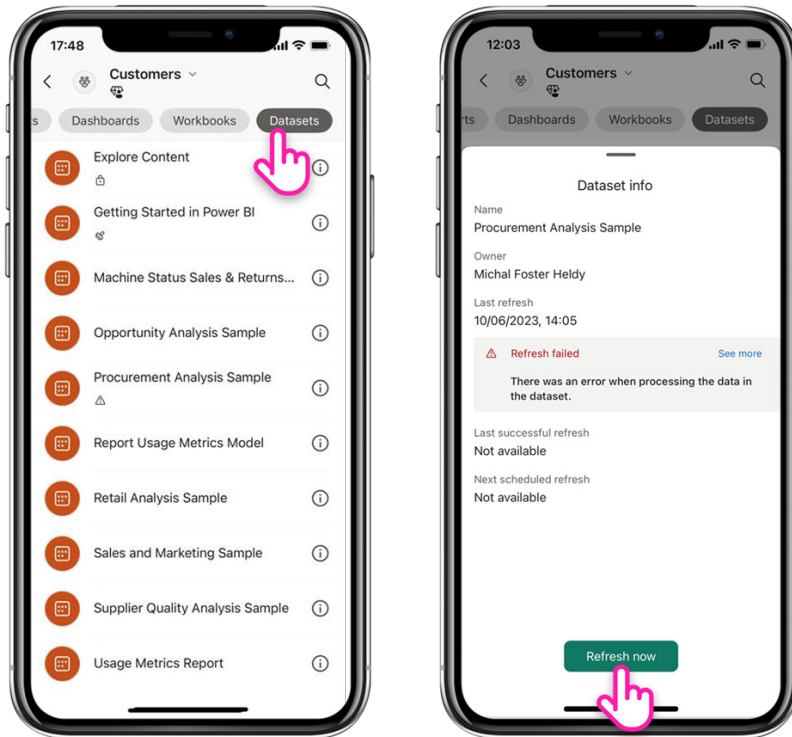
Datasets are coming to Power BI Mobile Apps

In the next Power BI Mobile App release, Microsoft is adding a long-awaited feature that will help dataset owners and report creators to manage their dataset directly from their phone. That means you will be able to see in your mobile device datasets. Go to a workspace, make sure to select the 'dataset' pill at the top and get the list of datasets that you have access to in that workspace.

When tapping on a dataset, you will get the dataset metadata pane,

which includes the name, owner, sensitivity label and also the latest refresh status. From this pane you may also trigger a dataset refresh all directly from your mobile application.

Dataset owners will also get push notifications when the scheduled refresh fails. They will be able to view the failure details and be able to re-try the refresh while they are on-the-go.



Boost your Custom Visuals performance

Microsoft has announced techniques to improve the performance of custom visuals, discussing the performance improvements made in both visual rendering and load times. They have identified and addressed certain bottlenecks in the code, and these improvements are available for any visual that has been updated to API version 4.2 and onwards. Along with these fixes, Microsoft has also provided code practices and techniques that can greatly enhance the performance of rendering custom visuals. It is recognised that getting the visual to render as quickly as possible is critical when writing the code.

Having said all this, as the Power BI platform continues to be improved and enhanced, new versions of the API will constantly be released. In order to get the most out of the Power BI visuals' platform and feature set, it is recommended that you keep up-to-date with the most recent version. To find out which version you're using, check the **aversion** in the **pbviz.json** file.

In the meantime, here are some recommendations for achieving optimal performance for your custom visual.

REDUCE PLUGIN SIZE

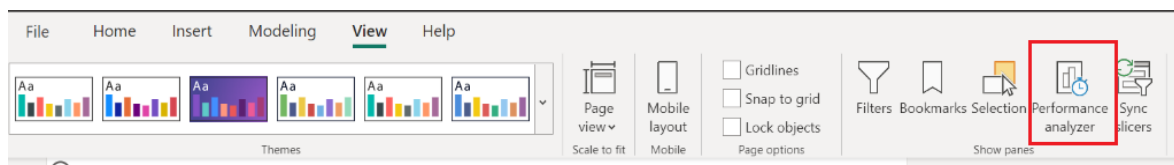
A smaller custom visual plugin size results in:

- faster download times
- faster installation whenever the visual is run.

CHECK RENDER TIME OF THE VISUAL

Measure the render time of your visual in various situations to see which, if any, parts of the script require optimisation.

POWER BI DESKTOP PERFORMANCE ANALYZER



Use the Power BI Desktop Performance Analyzer (*sic*) (**View -> Performance Analyzer**) to check how your visual renders in the following cases:

- first render of the visual
- thousands of data points
- a single data point / measure (to determine the visual render overhead)
- filtering
- slicing
- resizing (this may not work in the Performance Analcczer).

If possible, compare these measurements with those of a similar core visual to see if there are parts that may be optimised.

USE THE USER TIMING API

Use the User Timing API to measure your app's JavaScript performance. This API can also help you decide which parts of the script need optimisation.

OTHER WAYS TO TEST YOUR CUSTOM VISUAL

There are various ways to test you custom visual:

- code instrumentation: use the following console third party tools to gather data about your custom visual's performance:
 - `console.log()`
 - `console.dir()`
 - `console.time()`
 - `console.timeEnd()`
- the following web developer tools may also help measure your visual's performance, but keep in mind that they profile Power BI as well:
 - Metrics
 - JavaScript profiler.

Once you determined which parts of your visual need optimisation, you can then implement these further tips.

UPDATE MESSAGES

When you update the visual:

- don't re-render the entire visual if only some elements have changed. Render only the necessary elements
- store the data view passed on update. Render only the data points that are different from the previous data view. If they haven't changed, there's no need to re-render them
- resizing is often done automatically by the browser and doesn't require an update to the visual.

CACHE DOM NODES

When a node or list of nodes is retrieved from the DOM (Data Object Model), think about whether you can reuse them in later computations (sometimes even the next loop iteration). As long as you don't need to add or delete more nodes in the relevant area, caching them can improve the application's overall efficiency.

To make sure that your code is fast and doesn't slow down the browser, keep DOM access to a minimum. For example, instead of:

JavaScript

```
public update(options: VisualUpdateOptions) {  
    let axis = $(".axis");  
}
```

try:

JavaScript

```
public constructor(options: VisualConstructorOptions) {  
    this.$root = $(options.element);  
    this.xAxis = this.$root.find(".xAxis");  
}  
  
public update(options: VisualUpdateOptions) {  
    let axis = this.axis;  
}
```

AVOID DOM MANIPULATION

Limit DOM manipulations as much as possible. Insert operations like **prepend()**, **append()** and **after()** are time-consuming and should only be used when necessary. For example, instead of:

JavaScript

```
for (let i=0; i<1000; i++) {  
    $('#list').append('<li>'+i+'</li>');  
}
```

try making the above example faster by using **html()** and building the list beforehand:

JavaScript

```
let list = '';  
for (let i=0; i<1000; i++) {  
    list += '<li>'+i+'</li>';  
}  
$('#list').html(list);
```

RECONSIDER JQUERY

Limit JS frameworks and use native JS whenever possible to increase the available bandwidth and lower your processing overhead. Doing this might also decrease compatibility issues with older browsers.

ANIMATION OPTIONS

For repeated use of animations, consider using Canvas or WebGL instead of SVG. Unlike SVG, with these options, performance is determined by size rather than content.

CANVAS PERFORMANCE TIPS

You should read up on third party sites for tips on improving canvas performance, such as"

- faster load times
- improving HTML5 Canvas performance
- optimising canvas generally.

ANIMATION FUNCTIONS

Use **requestAnimationFrame** to update your on-screen animations, so your animation functions are called before the browser calls another repaint.

ANIMATION LOOPS

Does the animation loop redraw unchanged elements? If so, it wastes time drawing elements that don't change from frame-to-frame. Instead, you should update the frames selectively.

When you're animating static visualisations, it's tempting to lump all the draw code into one update function and repeatedly call it with new

data for each iteration of the animation loop. Instead, consider using a visual constructor method to draw everything static. Then, the update function only needs to draw visualisation elements that change.

In particular, Microsoft has noted that inefficient animation loops are often found in axes and legends.

Common issues

- Text size calculation: when there are a lot of data points, don't waste time calculating text size for each point. Calculate a few points and then estimate
- If some elements of the visual aren't seen in the display, there's not need to render them.

New visuals in AppSource

Yet another raft for this month:

- Spider Chart for Power BI by VisioChart
- HTML Content (Lite)
- Stacked Lipstick Bar Chart (Standard)
- Stacked Lipstick Column Chart (Standard)
- 100% Stacked Column Chart with Values instead of % (Standard)
- Dual Axis Scatter Chart (Standard)
- Category Comparison Bar Chart
- Stacked Column with Percentage and Total in Label (Standard)
- Likert Scale Chart for Power BI by ChartExpo.

Elevate Your Map Charts with Drill Down Map PRO

Drill Down Map PRO by ZoomCharts is a custom map visual for Power BI that lets you show your data on an interactive map and give it location-based context.

Features include:

- **built-in shape layers:** use preset shapes for easy filtering of countries
- **custom shape layer support:** provide custom shapes through KML and GeoJSON files
- **lasso tool:** draw and save your own filter shapes on top of the map
- **node clustering capabilities:** clusters can be turned into Do(ugh)nut or Pie charts for category display
- **map base layer:** lets you choose from four [4] options:
 - Azure maps
 - Custom (OpenStreetMaps, Google, CartoDB etc.)
 - Image (e.g. floor plans)
 - None (visualise shapes without a background)
- **aura, image and custom label support.**

Popular use cases:

- **production:** monitoring production data by location
- **sales and marketing:** mapping sales results by region
- **public sector:** visualising environmental and sociodemographic data.\

The screenshot displays the 'Drill Down Map PRO' interface. At the top left is a logo with a gear and 'PRO' text. To its right is the title 'Drill Down Map PRO'. Further right is a logo for 'Drill Down Visuals FOR POWER BI'. Below the title, there are four panels illustrating features:

- Turn node clusters into donut charts:** A map of the United States with several circular donut charts overlaid on different locations.
- Filter with the lasso tool:** A map of the United States with a pink lasso shape drawn over a region, and several purple circular nodes within that region.
- Change base layer:** A floor plan image with several colored circles (yellow, green, pink) overlaid on different rooms.
- Add auras:** A map of the United States with several large, semi-transparent colored circles (purple, pink, green) overlaid on different regions, each containing a house icon.

At the bottom center of the screenshot, the URL 'zoomcharts.com/powerbi' is visible.

"All-in-one" Multi Target KPI card

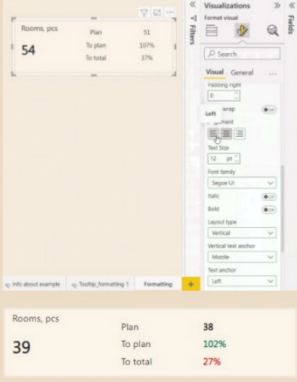
The Multi Target KPI card works with a single query and includes three additional indications, multiple categories, a pixel perfect alignment setting and built-in conditional formatting. You can change the settings of layout type and colour conditional formatting for additional measures in the visual and it is fairly simple for non-designers to use too.

Simply select the desired measure and category, if necessary. Add up to three [3] additional indicators to provide the context you need for your metric. It will help you improve reporting performance and save you time when designing and developing supplemental measures.

Navigation

- Main
- About card
- Formatting
- Design Light
- Design Dark
- Example
- Support


Customize the card to meet your demand



- 1 Add the necessary measures; the main and additional measures, a category and tooltips;
- 2 Enter the width percentage value, alignment, and color formatting for the main measure;
- 3 Decide to on/off the display of category names;
- 4 Choose a background and border color, transparency;
- 5 For the category, specify the number of cards per row and space between cards;
- 6 In additional measures, specify the parameters for the names and values separately.

Rooms, pcs	Plan	38
39	To plan	102%
	To total	27%

© Institute of Business Intelligence | 2022




Multi target KPI

by Institute of Business Intelligence

Crisp-n-clear visualization for your KPIs!

Also specific labels alignment, which is possible with separate text labels. Instead of this we developed «all-in-one» KPI card and share it with you for free.



WebView2 now Generally Available

WebView2 is now Generally Available. I think we might have said that in the heading..?

More next month!

New Features for Excel

The latest updates saw quite a few highlights. There is 'Formula Suggestions' and 'Formula by Example' for Excel web users. 'Use Image' and Data Types in PivotTables are now available for Windows and Mac Insider users, and 'Generate an Accessible PDF' is available to all Mac users.

You can check out the full list here:

Excel for the web

- Formula Suggestions
- Formula by Example
- Support Chart Formatting via Selection and Ribbon Buttons
- Move Chart Elements within Chart Player
- Name Manager

Excel for Windows

- Office Scripts (Current Channel and Monthly Enterprise Channel)
- Get data from Dynamic Arrays (Current Channel)
- Rapid Refresh (Current Channel)
- Guided Reapply (Current Channel)
- Use Image and Data Types in PivotTables (Insiders)
- Data Validation Dropdown List Autocomplete (Current Channel)
- Paste Values directly into your workbook using a Keyboard Shortcut (Current Channel)

Excel for Mac

- Generate an Accessible PDF
- Enhanced Search Experience
- Use Image and Data Types in PivotTables (Insiders).

Let's get started.

Formula Suggestions

Back in December last year, Microsoft announced 'Formula Suggestions' was beginning to roll out to Excel web users. Now, it is available to all web users with English set as their primary language (rules me out then: I use profane). Whether you're new to Excel or have used it for years, 'Formula Suggestions' helps you create formulae more quickly and accurately by offering relevant suggestions based on your data and context. All you need to do is type "=" at the beginning of a cell, and

Formula Suggestions will show you different formulae that you can apply to your data, along with a specified range.

'Formula Suggestions' can help you in several ways. First, it can improve your productivity by reducing the time and effort you need to type and edit formulae. For example, look at how Formula Suggestions suggests a simple summation here:

	A	B	C	D	E
1	Product	Q1 Sales			
2	Chains	2,940			
3	Socks	4,338			
4	Tights	4,194			
5	Shorts	1,829			
6	Tights	3,596			
7	Handlebars	4,132			
8	Socks	1,618			
9	Brakes	4,837			
10	Mountain Bikes	4,018			
11	Brakes	3,651			
12	Helmets	1,900			
13	Lights	2,697			
14	Locks	4,977			
15	Bottom Brackets	3,955			
16	Total	=SUM(B2:B15)			

A tooltip is shown below the formula cell, displaying a lightbulb icon, the formula `SUM(B2:B15) = 48,682`, a hand cursor, and a `Give feedback` link.

Second, 'Formula Suggestions' can provide you with robust formulas that will not break if you alter your data as it automatically adjusts the ranges to match your data structure. For example, totals are affected

when you add a new row to a table. If you use Formula Suggestions, the range will automatically update to include the new row, but if you manually add the cells, it will not – as in the following example:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1															
2		Manually					Formula Suggestions								
3		SumProduct team:					SumProduct team:								
4															
5			Guanting	10			Guanting	10							
6			Henrique	20			Henrique	20							
7			Kathryn	30			Kathryn	30							
8			Liam	40			Liam	40							
9			Myles	50			Myles	50							
10			Oscar	60			Oscar	60							
11			Sam	70			Sam	70							
12			Talia	80			Talia	80							
13			Tim	90			Tim	90							
14				450				=SUM(I5:I13)							
15															
16				=D5+D6+D7+D8+D9+D10+D11+D12+D13											

A tooltip is shown below the formula cell, displaying a lightbulb icon, the formula `SUM(I5:I13) = 450`, a hand cursor, and a `Give feedback` link. A list of suggested formulas is also visible: `SUBSTITUTE`, `SUBTOTAL`, `SUM`, `SUMIF`, `SUMIFS`, `SUMPRODUCT`, and `SUMSQ`.

Here, in cell **D14**, I have typed in the rather dreadful formula

=D5+D6+D7+D8+D9+D10+D11+D12+D13

However, in cell **I14**, I have started to type the formula in to prompt 'Formula Suggestions' and then used the formula recommended. This gives similar calculations:

	A	B	C	D	E	F	G	H	I	J
1										
2		Manually						Formula Suggestions		
3		SumProduct team:						SumProduct team:		
4										
5			Guanting	10				Guanting	10	
6			Henrique	20				Henrique	20	
7			Kathryn	30				Kathryn	30	
8			Liam	40				Liam	40	
9			Myles	50				Myles	50	
10			Oscar	60				Oscar	60	
11			Sam	70				Sam	70	
12			Talia	80				Talia	80	
13			Tim	90				Tim	90	
14				450					450	
15										
16				=D5+D6+D7+D8+D9+D10+D11+D12+D13					=SUM(I5:I13)	
17										

This will clearly give a better row should team members be added (rows inserted) or removed (rows deleted).

Third, 'Formula Suggestions' can help you discover new possibilities of what you can do with your data, by showing you different formulae that are relevant for your analysis. Whether you are a beginner or an expert in Excel, 'Formula Suggestions' can offer you a convenient and intuitive way to create formulae. 'Formula Suggestions' can suggest the following aggregation formulae:

- **SUM**
- **MIN**
- **MAX**
- **AVERAGE**
- **COUNT**
- **COUNTA**

These formulae are widely used in Excel to perform various calculations and analyses on data, such as finding the sum, minimum, maximum, average or count of values in a range.

For example, 'Formula Suggestions' may also assist in cell-counting tasks. The **COUNT** formula counts cells that contain numbers, whilst **COUNTA** will count all non-empty cells regardless of the type of data they contain. Using the correct formula is crucial to get the correct results, and this may be achieved by leveraging 'Formula Suggestions' ability to predict the correct formula for each of the use cases.

Formula Suggestions currently provides suggestions for basic aggregation formulae, helping users who are not yet acquainted with these Excel formulae. However, experienced users may benefit greatly from this feature as well, as it suggests not only the appropriate formula but also the appropriate range for that formula. The range suggestions can save a great deal of time, and can also prevent us from selecting incorrect ranges, which may lead to wrong results.

At the time of this article, Formula Suggestions is available to Excel web users that have English set as their primary language.

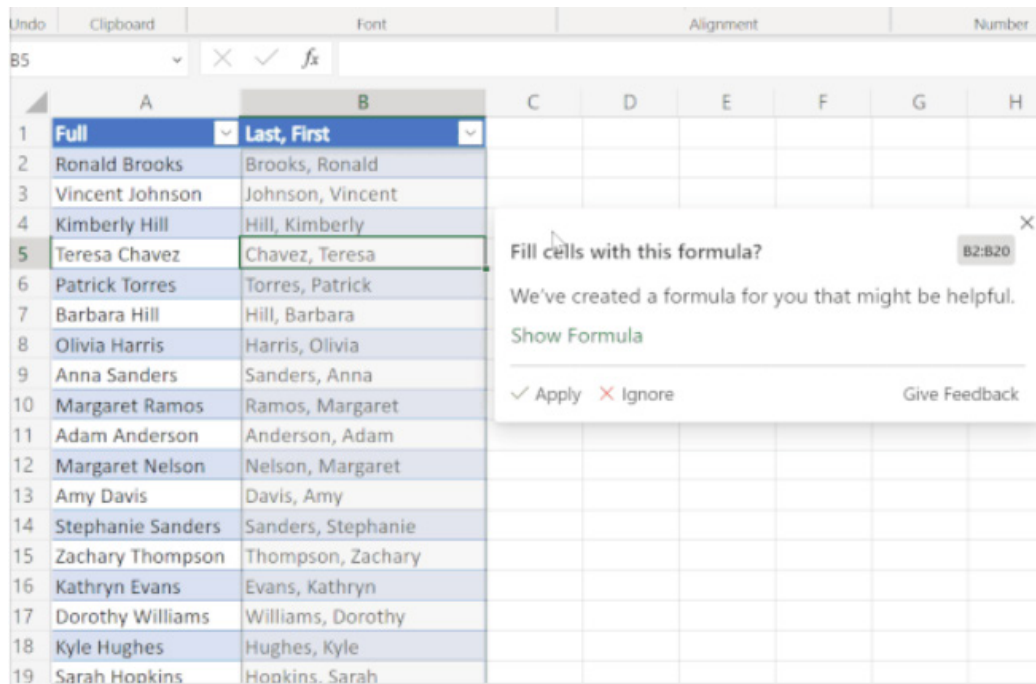
Formula by Example

Last month, 'Formula by Example' started to roll out to Excel web users. 'Formula by Example' looks for patterns as the user enters data in the worksheet. When it recognises a pattern, 'Formula by Example' offers a formula to fill the rest of the column with the recognised pattern.

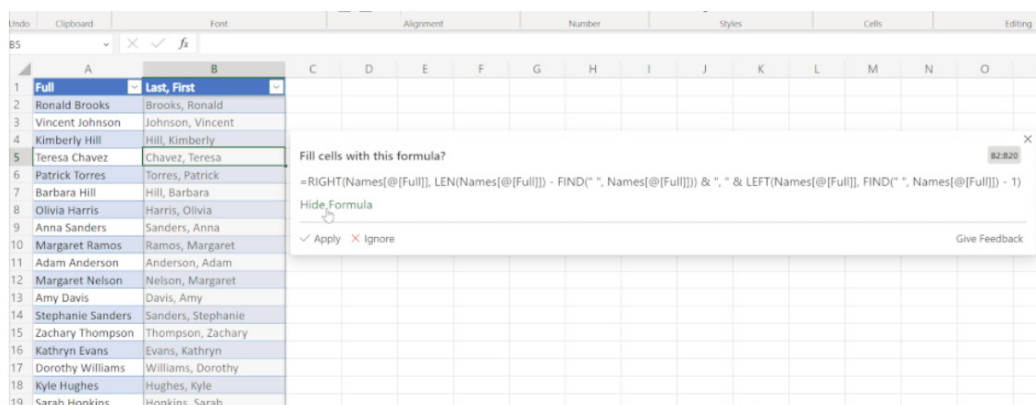
Consider 'Formula by Example' helping to reverse the first and last name order here, where we start typing in what we want:



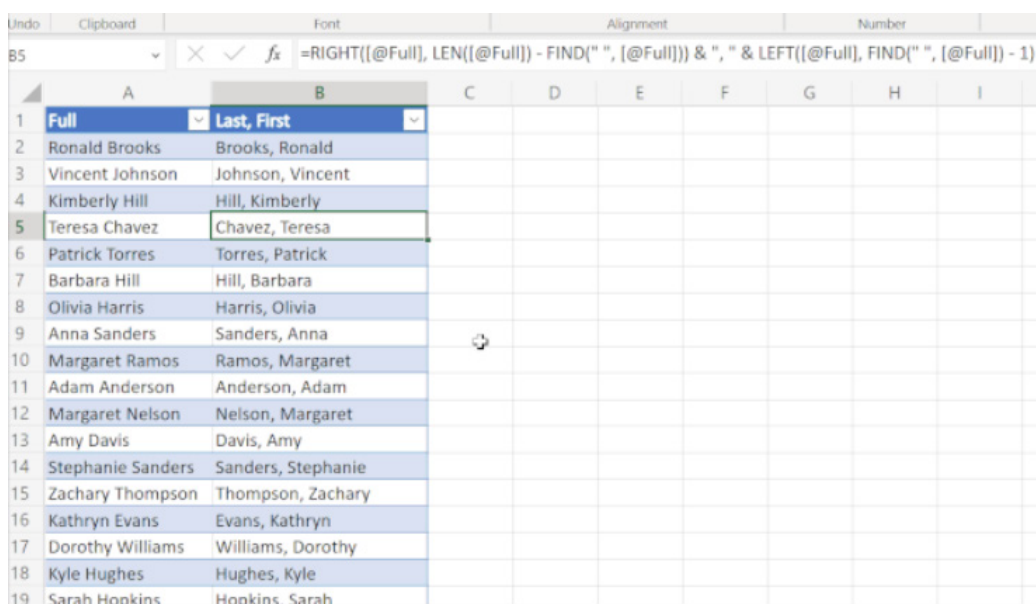
At this stage, 'Formula by Example' interrupts us:



We can click on 'Show Formula':



All you have to do now is 'Apply':



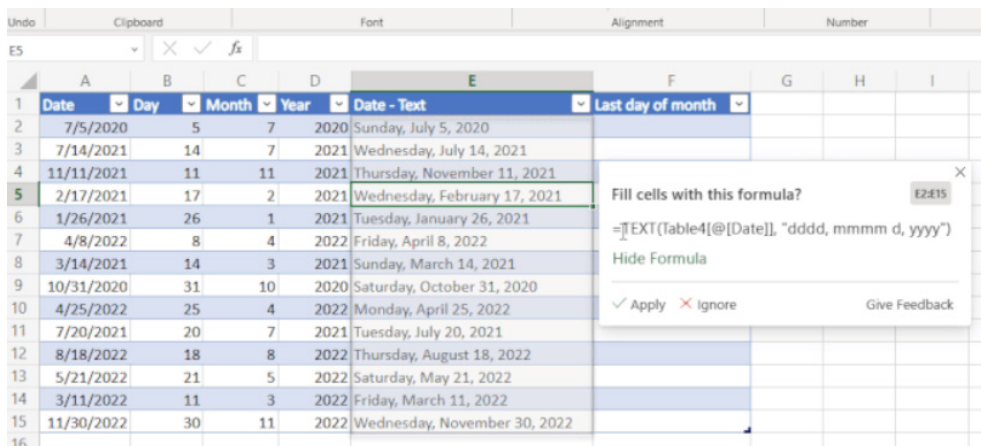
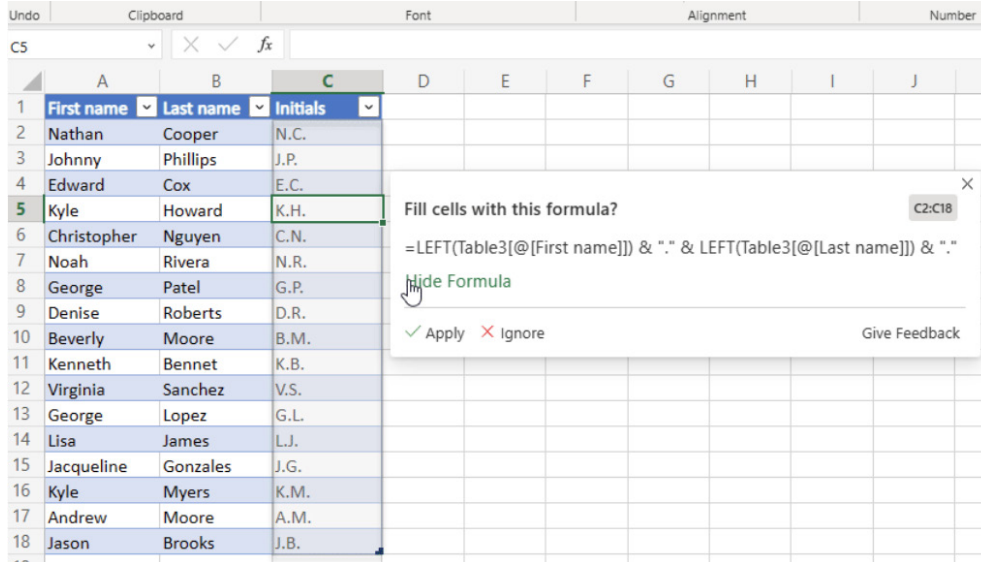
It's Flash Fill with formulae (more on that later)!

Currently, 'Formula by Example' supports Excel tables. Support for ranges is coming in a future release. In the meantime, if you wish to try 'Formula by Example' on a range of data today, convert it into a table (select range and click **Insert** -> **Table** or use the **CTRL + L** keyboard shortcut – **CTRL + T** does not work in Excel for the web).

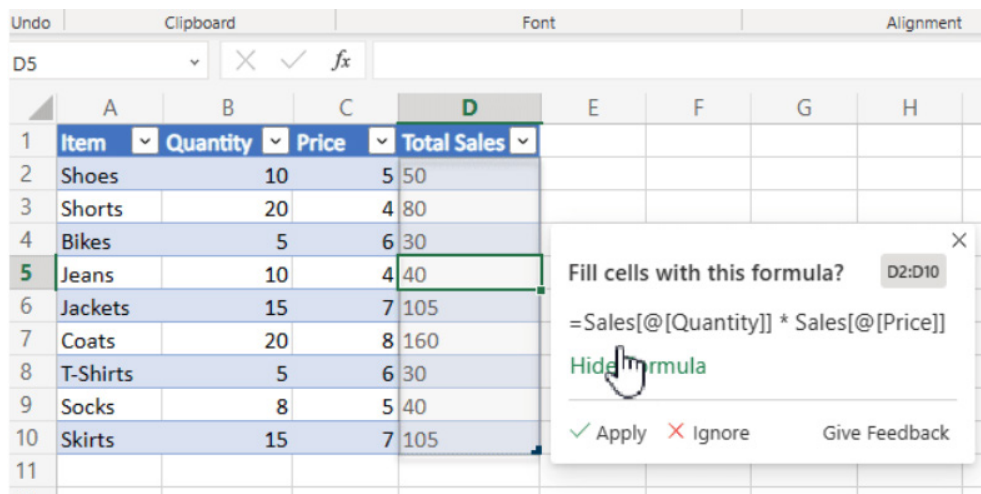
Once you have a "CTRL + L" Table, 'Formula by Example' suggestions will appear after you provide Excel with a few examples in a certain column.

Excel scans the column to identify a pattern in your data. When it finds a pattern, Excel will show a suggestion.

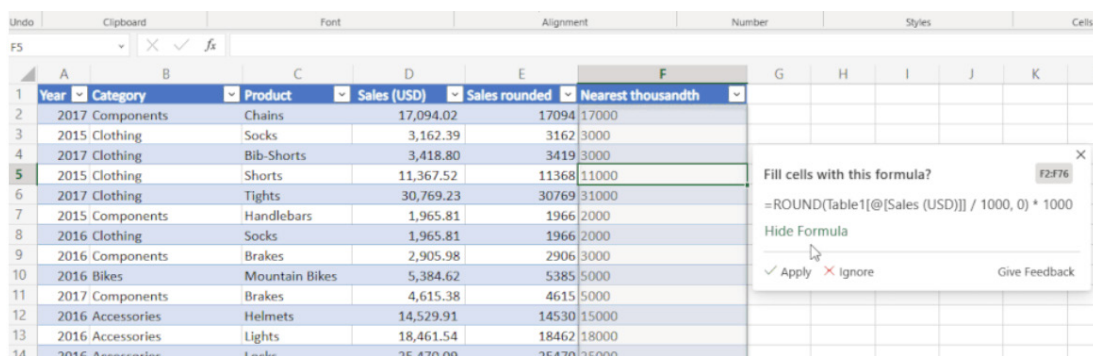
Formula by Example can recognise several patterns like text transformations, date transformations, arithmetic calculations, row numbering and forward filling. It's pretty simple:



'Formula by Example' will identify if you are trying to perform an arithmetic operation on different columns. Let's say we want to find the total sales by multiplying the Price by Quantity. By typing the first couple of results, Excel can complete the rest of the column with the arithmetic calculation formula:

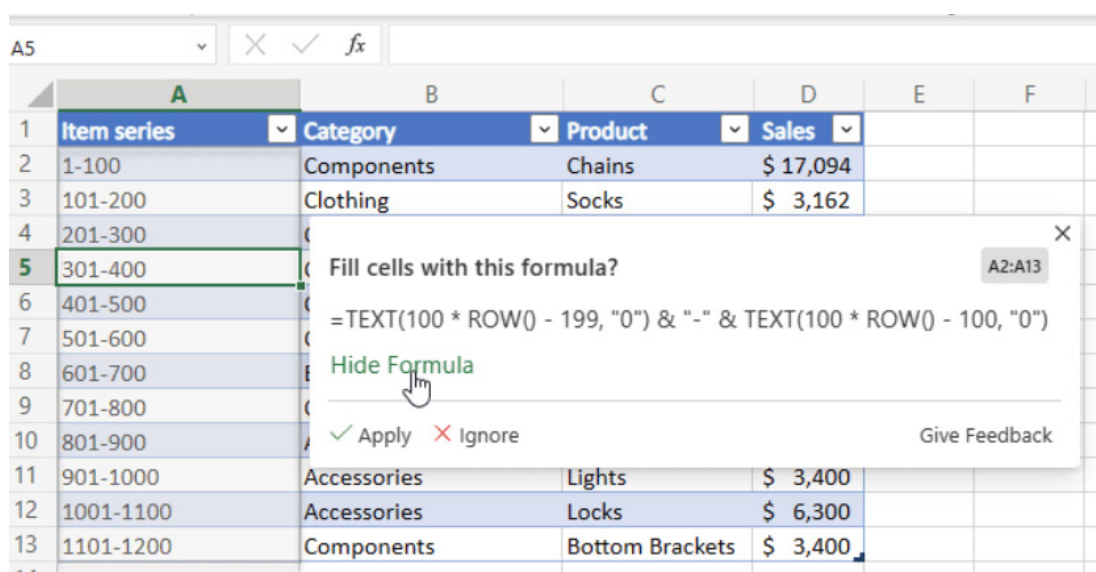


You can also use 'Formula by Example' to apply different types of rounding to your numbers.



'Formula by Example' allows you to create a dynamic row numbers column from example. This could come in handy in cases where you'd want your numbering to dynamically adjust if you add or remove a row.

You can also use Formula by Example to forward fill the rest of the column based on the first examples.



At the time of writing, 'Formula by Example' is available on Excel web for all US English users of OneDrive for Microsoft 365 Personal or Family. 'Formula by Example' is rolling out to users of SharePoint and OneDrive for Business. Additional language support will be available in a future release.

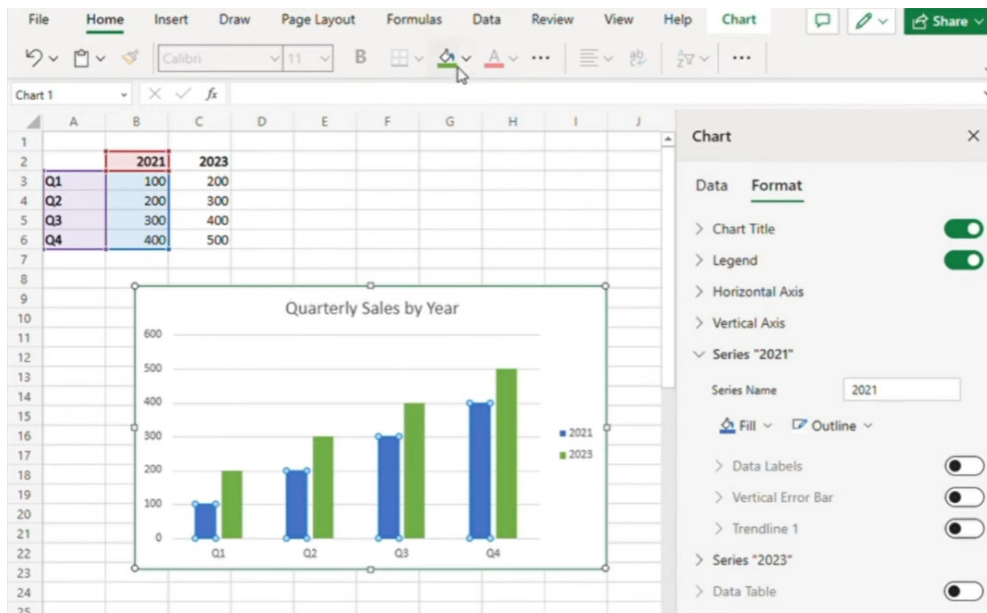
It should be noted that filling column data based upon a pattern is something that Excel has been able to do for many years, using a feature called Flash Fill. However, Flash Fill's suggestions are only provided as static text. If you wanted to change some of your input data or reuse the suggestion on different cells, you wouldn't be able to do so, because you wouldn't get a formula as an output. With 'Formula by Example', you will now see a formula that you can easily change, copy and reuse anywhere you want.

'Formula by Example' builds upon the technology that enables Flash Fill, leveraging logical-reasoning-based symbolic techniques to efficiently search for formulae that match the user-provided input and output examples. It then ranks the formulae to pick one that likely matches the user's intent – and if that's not the case, the user can provide additional representative examples to guide the tool.

'Formula by Example' is the next generation of the Flash Fill technology that covers a wider variety of transformations (including datetime and numeric manipulations) and generates readable Excel formulae. This formula-by-example technology has recently also been released in Power Automate and Power Apps.

Support Chart Formatting via Selection and Ribbon Buttons

You can now format your chart using the Home Ribbon Formatting controls in Excel for the web.



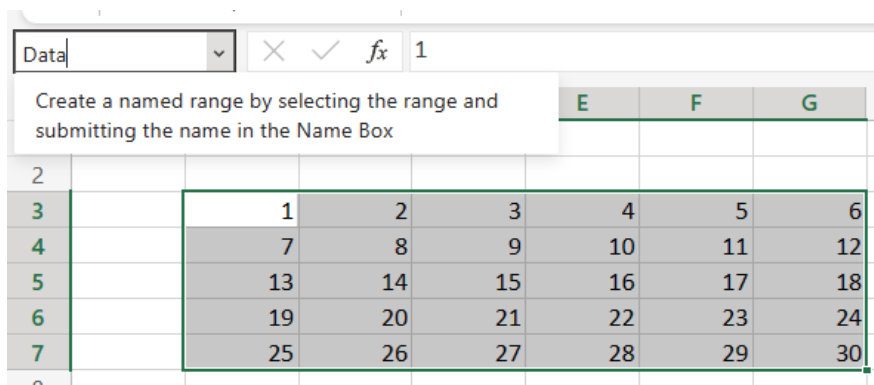
Move Chart Elements within Chart Player

You may now click and drag chart elements and the trend line equation to move them in a given chart in Excel for the web. This expands users' ability to customise how their chart looks.



Name Manager

You may now create named ranges from the Name Box in Excel for the web. Users may also search in the name box for any named ranges.



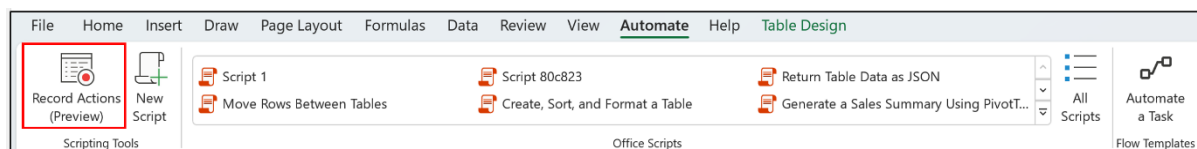
Office Scripts (Current Channel and Monthly Enterprise Channel)

In Excel for Windows, you now have the ability in the Current and Monthly Enterprise channels to record repetitive spreadsheet tasks and turn them into replayable Office Scripts.

Previously, you could create and modify automations using JavaScript and TypeScript using the Office Scripts Code Editor. With the Action Recorder, you can automate repetitive worksheet tasks without needing any programming experience.

How it works:

1. Open any existing workbook
2. Select **Automate -> Record Actions (Preview)**



3. Record actions in your workbook
4. When you're done recording, click the Stop recording button in the 'Record Actions (Preview)' task pane to save the recorded actions into a script that you can run at any time on any workbook.

Not all actions are recordable at this time. Microsoft is continuing to add support to increase the number of recordable actions. They encourage you to continue trying to record actions useful to your workflow as that will help the team prioritise the order in which they add support.

To use Office Scripts, you must have the following:

- a stable internet connection
- a commercial (E3/E5) or Education Microsoft 365 subscription
- Microsoft Edge WebView2 installed.

Get data from Dynamic Arrays (Current Channel)

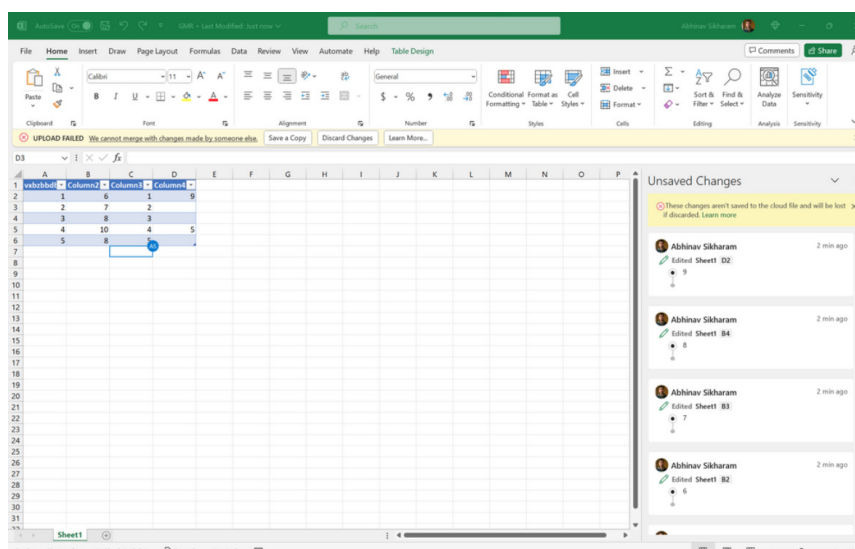
Also for Excel for Windows (Current Channel), 'Get Data from Table/Range' now includes Dynamic Arrays. You simply select any Dynamic Array on the grid and import it using Power Query.

Rapid Refresh (Current Channel)







'Rapid Refresh' is a feature that automatically reloads the workbook in co-authoring scenarios where the user may be prompted to refresh manually. Multi-workbook support and support in Excel for Mac will be coming shortly.

Guided Reapply (Current Channel)

In Excel for Windows previously, when co-authoring was interrupted for one reason or another, users would have to select 'Save As' (which creates another copy of the file) or else 'Discard' their changes to get back to the shared file. With 'Guided Reapply', users may review the pending changes and return to the shared file as they wish.



Use Image and Data Types in PivotTables (Insiders)

Sum of Sales (\$)		Country/Region		Grand Total
Category	Image			
Bookshelf		\$ 434,097	\$ 82,890	\$ 516,987
Armchair		\$ 387,932	\$ 279,825	\$ 667,757
Console table		\$ 338,252	\$ 221,312	\$ 559,564
Desk		\$ 108,000	\$ 208,376	\$ 316,376
Grand Total		\$ 1,268,281	\$ 792,403	\$2,060,684

PivotTables allow you to easily create summaries, dashboards and reports, as well as aggregate information. Whilst data types and in-cell images are a powerful part of modern Excel, PivotTables have only been able to use a text description of these modern content types. Now, the





















images and data types come intact to your PivotTable rows and columns in both Excel for Windows and Excel for Mac. They also include the same interactions you have in your source data (image cards, data type icons, data type cards, etc.).

To use it:

1. Create a PivotTable with images or data types in the source data
2. Add your fields with images or data types to the PivotTable rows or columns.

Country/Region	Flag	Sum of Sales
China	flag of China	575,690
Mexico	flag of Mexico	459,180
Canada	flag of Canada	446,620
Japan	flag of Japan	154,450
Germany	flag of Germany	150,410
Vietnam	flag of Vietnam	135,880
South Korea	flag of South Korea	120,860
India	flag of India	90,990
Ireland	flag of Ireland	
Italy	flag of Italy	

Before

Country/Region	Flag	Sum of Sales
 China		575,690
 Mexico		459,180
 Canada		446,620
 Japan		154,450
 Germany		150,410
 Vietnam		135,880
 South Korea		120,860
 India		90,990
 Ireland		
 Italy		

Now

You can also paste the examples into Excel using the **IMAGE** function.

This feature is available to Insiders who are running:

- Windows: Version 2307 (Build 16609.20000) or later
- Mac: Version 16.74 (Build 23060401) or later.

Data Validation Dropdown List Autocomplete (Current Channel)

Dropdown lists are a handy way to make data entry and validation more efficient in Excel for Windows. Microsoft has now added AutoComplete functionality, which automatically compares the text typed in a cell to all

items in the dropdown list and displays only the items that match. You should now be able to spend less time scrolling through lists, dealing with data validation errors or writing complex code to handle this task. Yay!

Paste Values directly into your workbook using a Keyboard Shortcut (Current Channel)

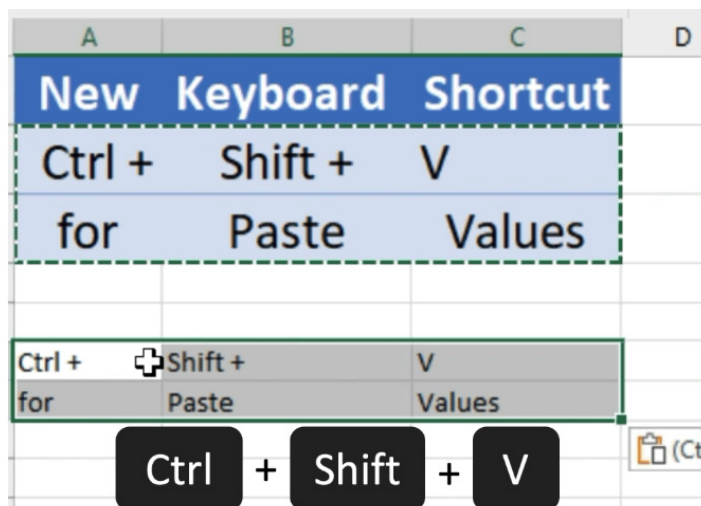
It's common to copy text from a web page, an email or another document, and then paste it into Excel (for Windows). Often, your goal is to match the format that is already in place in the cell or range where you're pasting. Until now, it was supposedly not possible to paste as plain text using shortcut keys (although I have been doing it using **ALT + E + S + Arrow keys**).

Likewise, when you copy a cell or range, and then want to paste values, you had a few options, but they all involved using multiple steps. You could:

- paste the copied content by pressing **CTRL + V**, and then select 'Paste Values' or 'Match Destination Formatting' in the 'Paste Options' dialog box
- select 'Paste Values' or 'Match Destination Formatting' in the Paste drop-down menu in the Ribbon (**ALT + H + V**)
- press **CTRL + ALT + V** to open the 'Paste Special' dialog, choose Values or Text, and then press **ENTER**.

Now you can simply press **CTRL + SHIFT + V**!

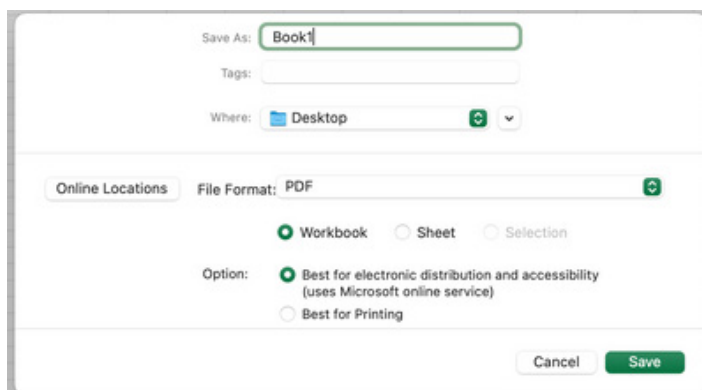
Depending on what you copied, this will either paste with 'Match Destination Formatting' or it will paste values.



This feature is rolling out to Beta Channel users running Version 2210 (Build 15726.20000) or later in Excel for Windows, and will be coming soon to Excel for Mac.

Generate an Accessible PDF

Previously when you exported to PDF in Excel for Mac, there were no tags added to communicate the structure of the document so all of the content was jumbled together. Now, the PDF service automatically creates accessibility tags, which brings the experience on par with Excel for Windows.



Enhanced Search Experience

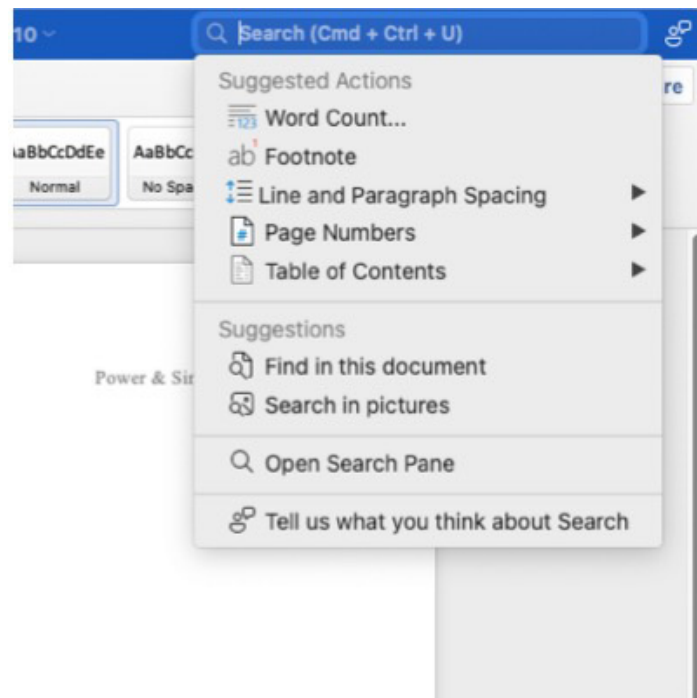
In Excel for Mac, the search experience has been revamped to provide a single place to find content from multiple sources. This function provides smart search suggestions before you type as well as supporting searching for more than just text in your document, *i.e.* you can also search for commands, help articles or web results (such as definitions, media and search engine results).

The new experience:

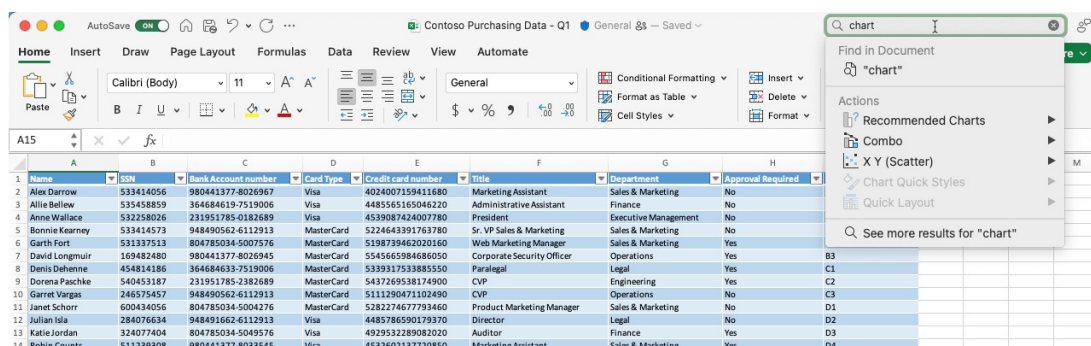
- provides smart search suggestions before you even type
- supports searching for more than just text in your document: you may also search for commands, help articles or web results
- makes the 'Smart Lookup' pane more accessible to people who are blind or have low vision with better support for VoiceOver as well as improved keyboard navigation within groups.

To access it:

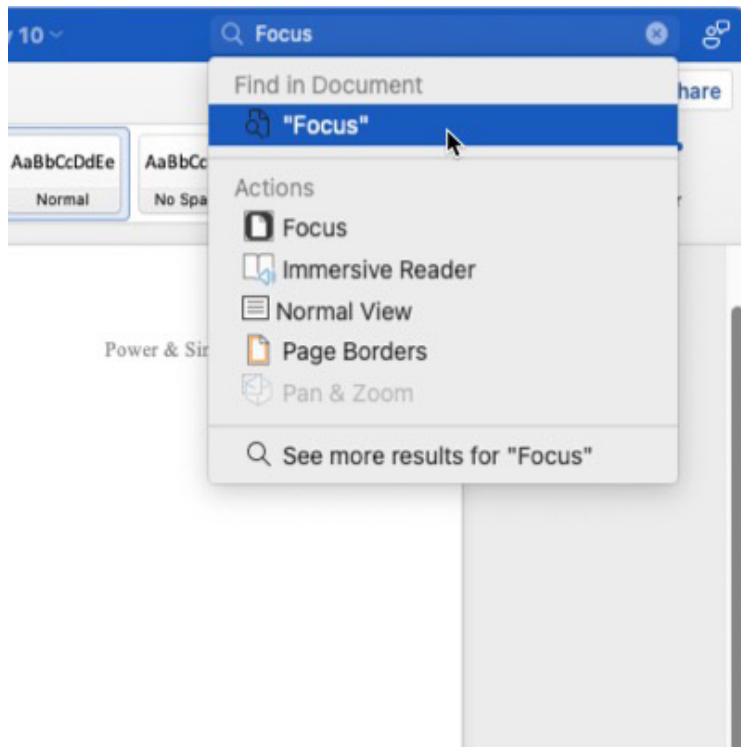
1. open an existing document, worksheet or presentation
2. click the Search box in the top-right corner of the app window, and select search suggestions



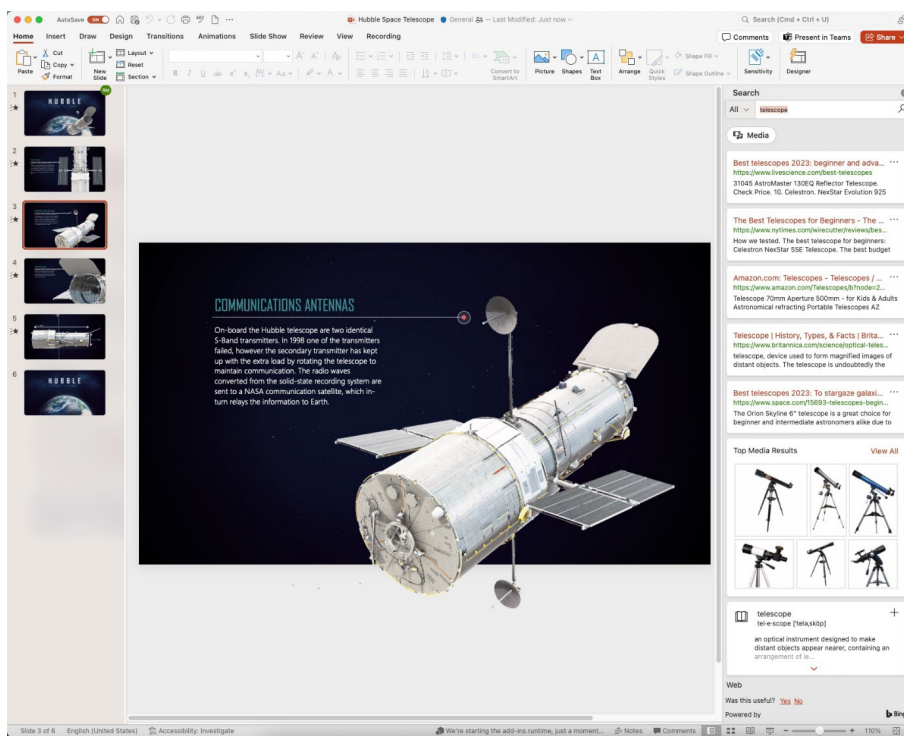
3. in the Search box, type a keyword or phrase you want to find in the document, worksheet or presentation
4. to search for a command, type the command name in the Search box



5. to search in the document, click 'Find in this document', and then enter the word or words you want to find



- to look for web or media results, click the 'Open Search' pane, or right-click anywhere on the document, worksheet or presentation, and then click the 'Open Search' pane in the shortcut menu.



If you prefer to use your keyboard, use the shortcuts to access the enhanced search experience:

Command	Shortcut
'Open Search' dropdown	CMD + CTRL + U
Find in document	CMD + F
'Open Search' pane	CMD + CTRL + L

It should be noted that the Search drop-down Excel menu does not close when you click the Ribbon tabs.

This feature is available to Beta Channel users who are running Version 16.74 (Build 23043001) or later, or to any users who have a Microsoft Office 2021 license.

The updated version of the grid with all the new features is fast becoming too complicated to show clearly here. Nonetheless, you can find the interactive links at aka.ms/ExcelFeaturesFlyer.

Excel Features Availability

Page 1 of 4

Feature	Insider		Production				Web
	Windows Find the latest Excel version for this platform	Mac Find the latest Excel version for this platform	Windows/CC Find the latest Excel version for this platform	Windows/MEC Find the latest Excel version for this platform	Windows/SA Find the latest Excel version for this platform	Mac Find the latest Excel version for this platform	
Block XL Add-Ins	Version 2302 (Build 16130.20128) or later						
PivotTables: Manual Sort of Rows & Columns			Already Supported	Already Supported	Already Supported	Already Supported	February 2023
Automatic Recalculation Optimization	Version 2208 (Build 15528.10000) or later	Version 16.64 (Build 22081401) or later					
Import Data from SQL Server Database		Version 16.68 (Build 22110801) or later					
Power Query Editor							
IMAGE function			Version 2211 (Build 15831.20190) or later	Version 2211 (Build 15831.20252) or later			December 2022
Check Formula with Value Preview Tooltips	Version 2302 (Build 16116.20000) or later	Version 16.70 (Build 230116) or later					
Office Scripts	Version 2212 (Build 15922.20000) or later	Version 16.68 (Build 22120161) or later					
Automate Tasks with Power Automate Tab			Version 2301 (Build 15703.10000) or later			Version 16.66 (Build 22092500) or later	
PivotTable Show Details to XLO							January 2023
Excel Live in Teams							December 2022
Formula Suggestions							December 2022*
Formula by Example							December 2022*
Suggested Links							December 2022
Add search bar in queries pane							December 2022

*Starting to roll out

Features Flyer: aka.ms/ExcelFeaturesFlyer

CC: Current Channel MEC: Monthly Enterprise Channel SA: Semi Annual Enterprise Channel All information is subject to change.

Excel Features Availability

Page 2 of 4

Feature	Insider		Production				Web
	Windows Find the latest Excel version for this platform	Mac Find the latest Excel version for this platform	Windows/CC Find the latest Excel version for this platform	Windows/MEC Find the latest Excel version for this platform	Windows/SA Find the latest Excel version for this platform	Mac Find the latest Excel version for this platform	
Add keyboard shortcut to open PQ editor			Version 2211 (Build 15730.31883) or later				
Create nested PQ data types	Version 2211 (Build 15928.10000) or later						
Add Get Data from Dynamic Arrays	Version 2105 (Build 14014.20002) or later						
Data from picture			Version 2210 (Build 15723) or later	Version 2210 (Build 15726.20262) or later			December 2022
Chart Data Folios							November 2022
Show Changes			Version 2209 (Build 15703.10000) or later				Version 16.66 (Build 22092500) or later
New Paste Options	Version 2210 (Build 15726.20000) or later						
Quickly Find the Command you need			Version 2206 (Build 15331.20010) or later				October 2022
New DAX Functions	Version 2208 (Build 15504.10000) or later						
Navigation Pane			Version 2209 (Build 15628.10000) or later				
Smooth Scrolling			Version 2209 (Build 15225.20092) or later	Version 2208 (Build 15601.20230)	Version 2208 (Build 15601.20456) or later	Already Supported	Already Supported
Check Performance							September 2022
Share Section of Excel Workbook							September 2022
Dynamic Array Support in Charts	Version 2209 (Build 15617.10000) or later			Version 2210 (Build 15726.20262) or later			September 2022
Modern Comments			Version 2209 (Build 15427.20000) or later				
Manage Your Storage Accounts from Mac		Version 16.64 (Build 22082100) or later					

Features Flyer: aka.ms/ExcelFeaturesFlyer

CC: Current Channel MEC: Monthly Enterprise Channel SA: Semi Annual Enterprise Channel All information is subject to change.

Excel Features Availability

Page 3 of 4

Feature	Insider		Production				Web
	Windows Find the latest Excel version for this platform	Mac Find the latest Excel version for this platform	Windows/CC Find the latest Excel version for this platform	Windows/MEC Find the latest Excel version for this platform	Windows/SA Find the latest Excel version for this platform	Mac Find the latest Excel version for this platform	
New Excel functions			Version 2208 (Build 15427.20194) or later	Version 2208 (Build 15601.20230) or later			Version 16.64 (Build 22081401) or later
Power Query Group operations							August 2022
Improvements to the connected Power BI experience	Version 2208 (Build 15601.20028) or later						August 2022
Add and edit rich text formatting			Already Supported	Already Supported	Already Supported	Already Supported	August 2022
Sort by color or icon from auto filter menu			Already Supported	Already Supported	Already Supported	Already Supported	August 2022
Edit files with legacy data connections			Already Supported	Already Supported	Already Supported	Already Supported	August 2022
Edit files with legacy Shared Workbook feature			Already Supported	Already Supported	Already Supported	Already Supported	August 2022
Delete chart elements							August 2022
Multiline formula bar							August 2022
Search within PivotTable Field List			Already Supported	Already Supported	Already Supported	Already Supported	July 2022
Set automatic data conversions	Version 2207 (Build 15427.20000) or later						
Natural Language Query Improvements			Version 2206 (Build 15330.20230) or later	Version 2205 (Build 15225.20354) or later			Version 16.63 (Build 22070801) or later
Resize Conditional Formatting dialog box		Version 16.64 (Build 22070600) or later					
Sheet protection			Already Supported	Already Supported	Already Supported	Already Supported	June 2022

Features Flyer: aka.ms/ExcelFeaturesFlyer

CC: Current Channel MEC: Monthly Enterprise Channel SA: Semi Annual Enterprise Channel All information is subject to change.

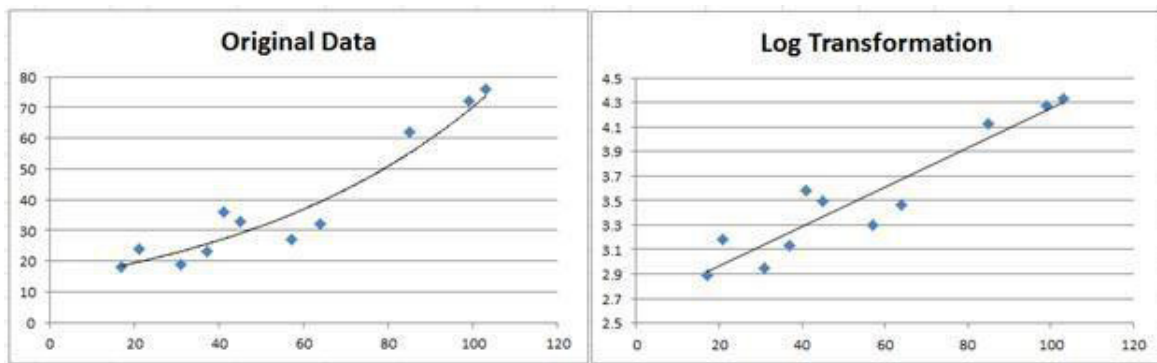
Feature	Insider		Production				Web
	Windows Find the latest Excel version for this platform	Mac Find the latest Excel version for this platform	Windows/CC Find the latest Excel version for this platform	Windows/MEC Find the latest Excel version for this platform	Windows/SA Find the latest Excel version for this platform	Mac Find the latest Excel version for this platform	
Semi-select for links creation			Already Supported	Already Supported	Already Supported	Already Supported	June 2022
Add "PivotTable Connections to Slicer settings pane"			Already Supported	Already Supported	Already Supported	Already Supported	June 2022
Import from local text, CSV, and XLSX files						Version 16.57 (22011100) or later	
Provide automatic alt-text suggestions on charts and PivotCharts			Version 2205 (Build 15225.20288) or later	Version 2204 (Build 15128.20280) or later		Version 16.62 (22061100) or later	
Power Query refresh for selected data sources			Already Supported	Already Supported	Already Supported	Already Supported	May 2022
Changing source file for workbook links			Already Supported	Already Supported	Already Supported	Already Supported	May 2022
Improved Recommended PivotTable experience	Version 2204 (Build 15128.10000) or later						
Faster recalc on resource constrained devices		Version 16.62 (Build 22050804) or later	Version 2204 (Build 15128.20248) or later	Version 2204 (Build 15128.20280) or later			
Faster AutoFilter				Version 2204 (Build 15128.20248) or later	Version 2208 (Build 15601.20456) or later	Version 16.61 (22050700) or later	
Dataflow connector				Version 2203 (Build 15028.20248) or later			
Dataverse connector			Version 2204 (Build 15128.20170) or later				
Improved Find dialog and Find All						Version 16.60 (220410) or later	
LAMBDA Helper Functions			Version 2202 (Build 14931.20130) or later	Version 2202 (Build 14931.20274) or later	Version 2208 (Build 15601.20456) or later	Version 16.56 (Build 211211) or later	Already Supported

Features Flyer: aka.ms/ExcelFeaturesFlyer

© 2022 Microsoft Corporation. All rights reserved. Microsoft, Excel, and the Excel logo are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

More next month, we're sure.

The A to Z of Excel Functions: LOGEST



Sometimes, you wish to forecast what comes next in a sequence, *i.e.* make a forecast. There are various approaches you could use:

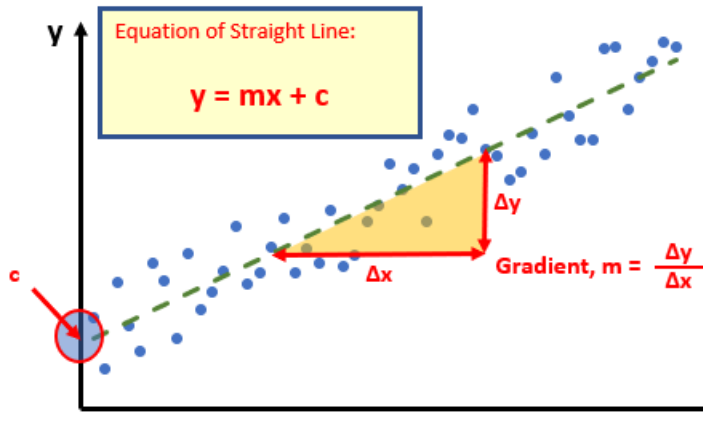
- **Naïve method:** this really does live up to its billing – you simply use the last number in the sequence, *e.g.* the continuation of the series 8, 17, 13, 15, 19, 14, ... would be 14, 14, 14, 14, ... Hmm, great
- **Simple average:** only a slightly better idea: here, you use the average of the historical 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, ...

- **Regression analysis:** 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:

For example, if you plotted your observable data on a chart, it might look something like this:

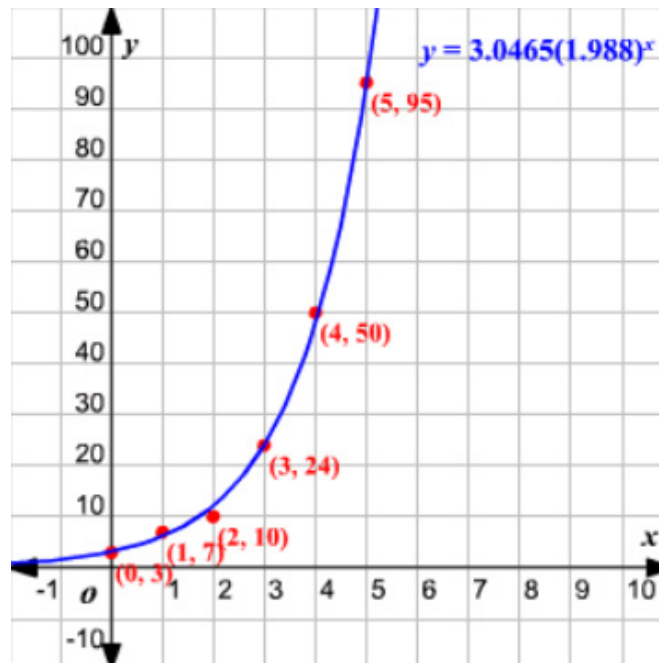


There is a statistical technique where you may actually draw the “best straight line” through the data using an approach such as Ordinary Least Squares. Once you have worked it out, you can calculate the gradient (**m**) and where the line cuts the **y** axis (the **y** intercept, **c**). This gives 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**

- **Exponential curve fitting:** if the relationship is not linear, a more suitable approach may be to calculate the best fit exponential curve:



The line may be defined by

$$y = bm^x$$

or in more complex situations,

$$y = bm_1^{x_1}m_2^{x_2}...$$

if there are multiple **x**-values, where the dependent **y**-value is a function of the independent **x**-values. The **m**-values are bases corresponding to each exponent **x**-value, and **b** is a constant value. Note that **y**, **x** and **m** can be vectors. The array that **LOGEST** returns is $\{m_n, m_{n-1}, \dots, m_1, b\}$.

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

$$\text{LOGEST}(\text{known_y's}, [\text{known_x's}], [\text{constant}], [\text{statistics}]).$$

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

- **known_y's**: this is required and represents the set of **y**-values that you already know in the relationship **y = bm^x** (nothing to do with bicycles)
- **known_x's**: this is optional and denotes the set of **x**-values that you may already know in the relationship **y = bm^x**
- **constant**: this argument is optional and is a logical value specifying whether to force the constant **b** to equal one (1)
- **statistics**: this final argument is also optional. This too is a logical value specifying whether to return additional regression statistics (see below).

It should be further noted that:

- if the range of **known_y's** is either in a single column or else a single row, each column of **known_x's** is interpreted as a separate variable
- the range of **known_x's** may include one or more sets of variables. If only one variable is used, **known_y's** and **known_x's** can be ranges of any shape, as long as they have equal dimensions. If more than one variable is used, **known_y's** must be a vector (*i.e.* a range with a height of one row or a width of one column)
- if **known_x's** is omitted, it is assumed to be the array {1,2,3,...} that is the same size as **known_y's**
- if **constant** is TRUE or omitted, **b** is calculated normally
- if **constant** is FALSE, **b** is set equal to one (1) and the **m**-values are adjusted to fit **y = m^x**
- if **statistics** is TRUE, **LOGEST** returns the additional regression statistics; as a result, the returned array is {**m_n, m_{n-1}, ..., m₁, c; se_n, se_{n-1}, ..., se₁, se_c; r², se_y; F, df; ss_{reg}, ss_{resid}**}
- if **statistics** is FALSE or omitted, **LOGEST** returns only the **m**-coefficients and the constant **b**.

With regard to the additional regression statistics, these are produced in a grid (an array) as follows:

	A	B	...	N-1	N	N+1
1	m _n	m _{n-1}	...	m ₂	m ₁	b
2	se _n	se _{n-1}	...	se ₂	se ₁	se _c
3	r ²	se _y				
4	F	df				
5	ss _{reg}	ss _{resid}				

These statistics may be described as follows:

Statistic	Description
se ₁ , se ₂ , ..., se _n	Standard error values for the coefficients m₁, m₂, ..., m_n . The standard error is a measure of the statistical accuracy of an estimate, equal to the standard deviation of the theoretical distribution of a large population of such estimates. It is usually estimated in practice as the sample standard deviation divided by the square root of the sample size (assuming statistical independence of the values in the sample), $\frac{\sigma}{\sqrt{n}}$
se _b	Standard error value for the constant c (but this is equal to #N/A when constant is FALSE)
r ²	This the coefficient of determination, which compares estimated and actual y-values and ranges, with a value between zero(0) and one (1). If it is 1, there is a perfect correlation in the sample, <i>i.e.</i> there is no difference between the estimated y-value and the actual y-value . At the other extreme, if the coefficient of determination is zero, the regression equation is not helpful in predicting a y-value . The coefficient of determination, R² , is the proportion of the variance in the dependent variable that is predictable from the independent variable(s). There are several definitions of R² , but they are not always equivalent (indeed, they can be negative on occasion). One class of such cases includes that of simple linear regression where r² is used instead of R² . When an intercept is included, then r² is simply the square of the sample correlation coefficient (<i>i.e.</i> r) between the observed outcomes and the observed predictor values. Here, the coefficient of determination will range between zero (0) and one (1)

Statistic	Description
se_y	Standard error for the y estimate
F	<p>This is the F statistic or the F-observed value. You should use the F statistic to determine whether the observed relationship between the dependent and independent variables occurs by chance</p> <p>This may be calculated using the F.INV.RT function in Excel. An F statistic is a value you get when you run an analysis of variance (ANOVA) test or a regression analysis to find out if the means between two populations are significantly different. It's similar to a T statistic from a T-Test: a T-test will tell you if a single variable is statistically significant, whereas an F test will tell you if a group of variables are jointly significant</p>
df	<p>The degrees of freedom. You should use the degrees of freedom to help you find F-critical values in a statistical table. Compare the values you find in the table to the F statistic returned by LOGEST to determine a confidence level for the model.</p> <p>The degree(s) of freedom is the number of independent values or quantities which may be assigned to a statistical distribution. It is calculated as follows.</p> <p>When no x columns are removed from the model due to collinearity:</p> <ul style="list-style-type: none"> if there are k columns of known_x's and constant = TRUE or is omitted, df = n - k - 1 if constant = FALSE, df = n - k. <p>In both cases, each x column that is removed due to collinearity increases the value of df by one (1)</p>
ss_{reg}	<p>This is the regression sum of squares</p> <p>In regression analysis, Excel calculates for each point the squared difference between the y-value estimated for that point and its actual y-value. The sum of these squared differences is called the residual sum of squares, ss_{resid}. Excel then calculates the total sum of squares, ss_{total}. When the constant argument = TRUE or is omitted, the total sum of squares is the sum of the squared differences between the actual y-values and the average of the y-values. When the constant argument = FALSE, the total sum of squares is the sum of the squares of the actual y-values (<i>without</i> subtracting the average y-value from each individual y-value). Then, the regression sum of squares, ss_{reg}, may be found from: $ss_{reg} = ss_{total} - ss_{resid}$. The smaller the residual sum of squares is, compared with the total sum of squares, the larger the value of the coefficient of determination, r^2, which is an indicator of how well the equation resulting from the regression analysis explains the relationship among the variables. The value of r^2 equals ss_{reg}/ss_{total}</p>
ss_{resid}	This is the residual sum of squares, as explained above.

To be clear:

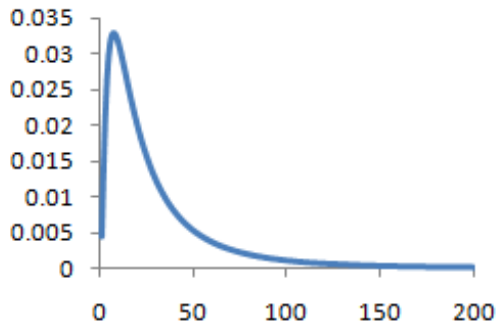
- the more a plot of your data resembles an exponential curve, the better the calculated line will fit your data. Like **LINEST**, **LOGEST** returns an array of values that describes a relationship among the values, but **LINEST** fits a straight line to your data; **LOGEST** fits an exponential curve
- when you have only one independent **x**-variable, you can obtain **y**-intercept (**b**) values directly by using the following formula:
y-intercept (b): INDEX(LOGEST(known_y's, known_x's), 2)
- you can use the **y = bm^x** equation to predict future values of **y**, but Excel has a **GROWTH** function to do this for you
- when entering an array constant such as **known_x's** as an argument, use commas to separate values in the same row and semicolons to separate rows. Separator characters may be different depending upon your regional settings
- you should note that the **y**-values predicted by the regression equation may not be valid if they are outside the range of **y**-values you used to determine the equation.

Please see our comprehensive example below:

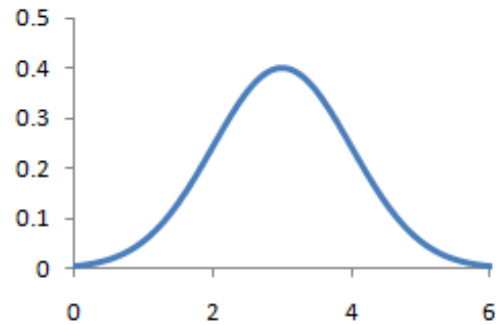
	A	B	C	D	E	F	G	H	I	
1	Floor Space	Offices	Entrances	Age	Assessed Value					
2	x_1	x_2	x_3	x_4	y		LOGEST			
3	2,310	2	2	20	\$142,000				=SE\$17*(SD\$17^A3)*(SC\$17^B3)*(SB\$17^C3)*(SA\$17^D3)	
4	2,333	2	2	12	\$144,000				\$141,966	
5	2,356	3	1.5	33	\$151,000				\$144,482	
6	2,379	3	2	43	\$150,000				\$150,595	
7	2,402	2	3	53	\$139,000				\$150,000	
8	2,425	4	2	23	\$169,000				\$138,888	
9	2,448	2	1.5	99	\$126,000				\$169,783	
10	2,471	2	2	34	\$142,900				\$126,071	
11	2,494	3	3	23	\$163,000				\$142,586	
12	2,517	4	4	55	\$169,000				\$161,179	
13	2,540	2	3	22	\$149,000				\$169,396	
14									\$149,942	
15										
16	Results					{=LOGEST(E3:E13,A3:D13,TRUE,TRUE)}				
17	0.9983	1.0181	1.0854	1.0002	80387.1472					
18	0.0001	0.0033	0.0025	0.0000	0.0758					
19	0.9972	0.0060	#N/A	#N/A	#N/A					
20	538.8904	6.0000	#N/A	#N/A	#N/A					
21	0	0	#N/A	#N/A	#N/A					
22										
23										
24	Statistics									
25	m_4	m_3	m_2	m_1	b					
26	se_4	se_3	se_2	se_1	se_c					
27	r^2	se_y								
28	F	df								
29	SS_{reg}	SS_{resid}								
30										

The A to Z of Excel Functions: LOGINV

Log normal

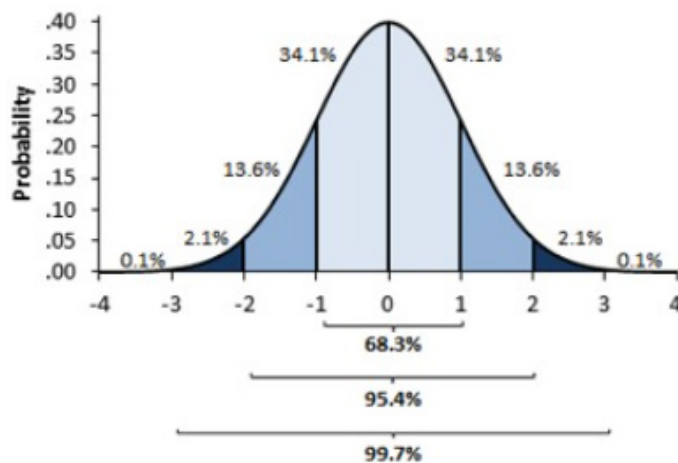


Normal



A Lognormal distribution is a statistical distribution of logarithmic values from an underlying Normal distribution. A Lognormal distribution may be translated to a Normal distribution and vice versa using the associated logarithmic translations.

A Normal distribution is a symmetrical probability distribution of outcomes that forms a bell curve. In a Normal distribution, 68% of the results fall within one standard deviation and 95% fall within two standard deviations, viz.



While most people are familiar with a Normal distribution, they may not be as familiar with the related Lognormal distribution. A Normal distribution may be converted to a Lognormal distribution using logarithms. It should be noted that Lognormal distributions can only arise from a Normally distributed set of random variables.

There can be a few reasons for using Lognormal distributions in conjunction with Normal distributions. Most commonly, Lognormal distributions are the result of taking the natural logarithm where the base is equal to $e=2.718...$. However, the Lognormal distribution may be scaled using a different base, if desired, which affects the shape of the Lognormal distribution.

The Lognormal distribution plots the logarithm of random variables from a Normal distribution curve. In general, the logarithm is known as the exponent to which a base number must be raised in order to produce the random variable (x) that is found along a Normally distributed curve. It should be noted that Lognormal distributions are positively skewed with long right tails due to low mean values and high variances in the random variables.

Clear as mud, yes?

In practice, Normal distributions may present a few problems that Lognormal distributions can solve. In particular, Normal distributions allow for negative random variables whilst Lognormal distributions include all positive variables.

Probably (*get it?*) the most common application where Lognormal distributions are used in finance is in the analysis of stock prices. A Lognormal distribution is more suitable for this purpose because asset prices cannot be negative. An important point to note is that when the continuously compounded returns of a stock follow a Normal distribution, then the stock prices follow a Lognormal distribution. The Lognormal distribution curve may therefore be used to help better identify the compound return that the stock can expect to achieve over a period of time.

The distribution is occasionally referred to as the Galton distribution or Galton's distribution, after Francis Galton. In summary, a Lognormal process is "simply" the statistical realisation of the multiplicative product of many independent random variables, each of which is positive and it is therefore the maximum entropy probability distribution for a random variate x for which the mean and variance of $\text{LN}(x)$ are specified. And if you follow all of that, get out of here because no one likes a smartarse.

The **LOGINV** function returns the inverse of the Lognormal cumulative distribution function of x , where $\text{LN}(x)$ is Normally distributed with parameters **mean** and **standard_deviation**.

If $p = \text{LOGNORMDIST}(x, \dots)$ then $\text{LOGINV}(p, \dots) = x$. It's confusing, as you might think $\text{LOGINV} = \text{EXP}$!

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

LOGINV(probability, mean, standard_deviation)

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

- **probability**: this is required and represents a probability associated with the Lognormal distribution
- **mean**: this is also required and denotes the mean of the natural logarithm of x , $\text{LN}(x)$
- **standard_deviation**: again required, this is the standard deviation of $\text{LN}(x)$.

It should be noted that:

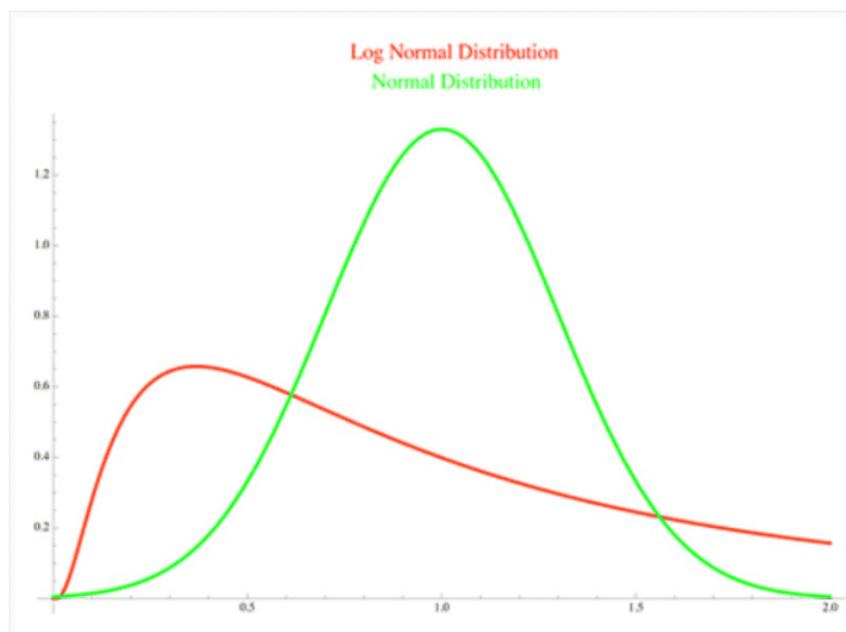
- this function has been replaced by **LOGNORM.INV**. Microsoft states that the new function may provide improved accuracy, as well as provide a function name that better reflects its usage
- although this function is still available for backward compatibility, consider using **LOGNORM.INV** from now on, because this function may not be available in future versions of Excel
- if any argument is nonnumeric, **LOGINV** returns the **#VALUE!** error value
- if **probability** ≤ 0 or **probability** ≥ 1 , **LOGINV** returns the **#NUM!** error value
- if $x \leq 0$ or if **standard_deviation** ≤ 0 , **LOGINV** returns the **#NUM!** error value
- the inverse of the lognormal distribution function is given by

$$\text{LOGINV}(p, \mu, \sigma) = e^{(\mu + \sigma \times \text{NORMSINV}(p))}$$

Please see the example below:

	A	B	C
1	Data	Description	
2	0.269704931	Probability associated with the lognormal distribution	
3	2	Arithmetic mean of $\text{LN}(x)$	
4	1	Standard deviation of $\text{LN}(x)$	
5			
6			
7	Formula	Description	Result
8	=LOGINV(A2,A3,A4)	Inverse of the lognormal cumulative distribution function for the terms above	4.000000

The A to Z of Excel Functions: LOGNORM.DIST



The **LOGNORM.DIST** function returns the Lognormal distribution of **x**, where **LN(x)** is Normally distributed with parameters **mean** and **standard_deviation**. The **LOGNORM.DIST** function employs the following syntax to operate:

LOGNORM.DIST(x, mean, standard_deviation, cumulative)

The **LOGNORM.DIST** function has the following arguments:

- **x**: this is required and represents the value at which to evaluate the function
- **mean**: this is also required and denotes the mean of the natural logarithm of **x**, **LN(x)**
- **standard_deviation**: again required, this is the standard deviation of **LN(x)**
- **cumulative**: yet again, this final argument is required. This is a logical value that determines the form of the function. If **cumulative** is **TRUE**, **LOGNORM.DIST** returns the cumulative distribution function; if **FALSE**, it returns the probability density function.

It should be noted that:

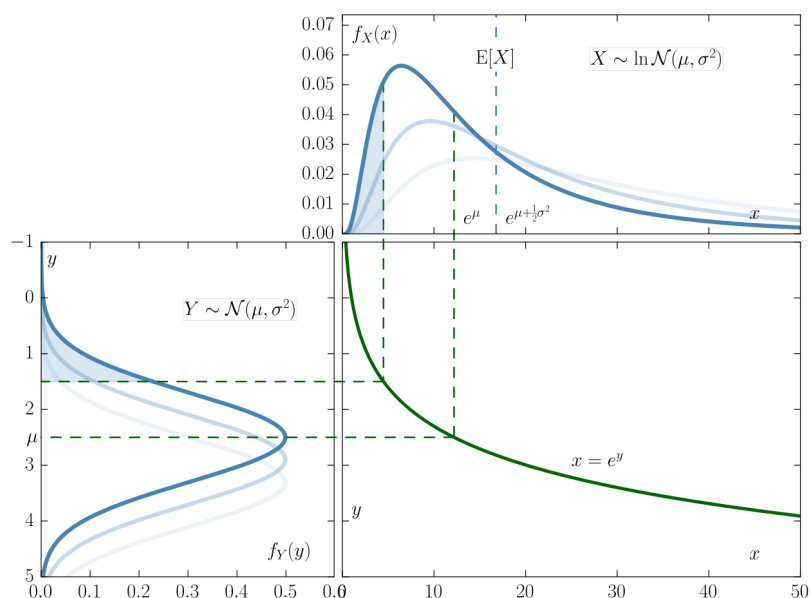
- if any argument is nonnumeric, **LOGNORM.DIST** returns the **#VALUE!** error value
- if **x** ≤ 0 or if **standard_deviation** ≤ 0, **LOGNORM.DIST** returns the **#NUM!** error value
- the equation for the lognormal cumulative distribution function is:

$$\text{LOGNORM.DIST}(x, \mu, \sigma) = \text{NORM.S.DIST}\left(\frac{\ln(x) - \mu}{\sigma}\right)$$

Please see our comprehensive example below:

	A	B	C
1	Data	Description	
2	4	Value for which you want the distribution	
3	2	Arithmetic mean of LN(x)	
4	1	Standard deviation of LN(x)	
5			
6			
7	Formula	Description	Result
8	=LOGNORM.DIST(A2,A3,A4,TRUE)	Cumulative lognormal distribution function for the terms above	0.269704931
9	=LOGNORM.DIST(A2,A3,A4,FALSE)	Probability lognormal distribution function for the terms above	0.082616141
10			

The A to Z of Excel Functions: LOGNORM.INV



The **LOGNORM.INV** function returns the inverse of the Lognormal cumulative distribution function of **x**, where **LN(x)** is Normally distributed with parameters **mean** and **standard_deviation**.

If **p = LOGNORM.DIST(x, ...)** then **LOGNORM.INV(p, ...) = x**.

The **LOGNORM.INV** function employs the following syntax to operate:

LOGNORM.INV(probability, mean, standard_deviation)

The **LOGNORM.INV** function has the following arguments:

- **probability**: this is required and represents a probability associated with the Lognormal distribution
- **mean**: this is also required and denotes the mean of the natural logarithm of **x**, **LN(x)**
- **standard_deviation**: again required, this is the standard deviation of **LN(x)**.

It should be noted that:

- this function has replaced **LOGINV**. Microsoft states that this new function may provide improved accuracy, as well as provide a function name that better reflects its usage
- although **LOGINV** is still available for backward compatibility, consider using **LOGNORM.INV** from now on, because the former function may not be available in future versions of Excel
- if any argument is nonnumeric, **LOGNORM.INV** returns the **#VALUE!** error value
- if **probability** ≤ 0 or **probability** ≥ 1 , **LOGNORM.INV** returns the **#NUM!** error value
- if **x** ≤ 0 or if **standard_deviation** ≤ 0 , **LOGNORM.INV** returns the **#NUM!** error value.

Please see our comprehensive example below:

	A	B	C
1	Data	Description	
2	0.269704931	Probability associated with the lognormal distribution	
3	2	Arithmetic mean of LN(x)	
4	1	Standard deviation of LN(x)	
5			
6			
7	Formula	Description	Result
8	=LOGNORM.INV(A2,A3,A4)	Inverse of the lognormal cumulative distribution function for the terms above	4.000000
9			

The **LOGNORM.INV** function returns the inverse of the Lognormal cumulative distribution function of **x**, where **LN(x)** is Normally distributed with parameters **mean** and **standard_deviation**.

If **p = LOGNORM.DIST(x, ...)** then **LOGNORM.INV(p, ...) = x**.

The **LOGNORM.INV** function employs the following syntax to operate:

LOGNORM.INV(probability, mean, standard_deviation)

The **LOGNORM.INV** function has the following arguments:

- **probability**: this is required and represents a probability associated with the Lognormal distribution
- **mean**: this is also required and denotes the mean of the natural logarithm of **x**, **LN(x)**
- **standard_deviation**: again required, this is the standard deviation of **LN(x)**.

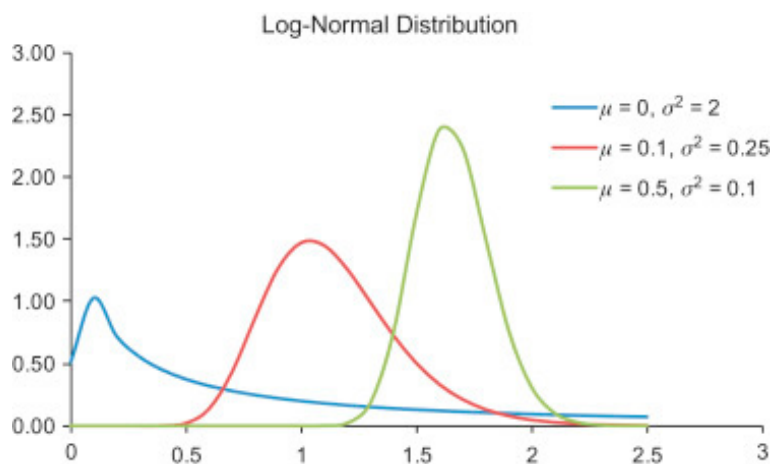
It should be noted that:

- this function has replaces **LOGINV**. Microsoft states that this new function may provide improved accuracy, as well as provide a function name that better reflects its usage
- although **LOGINV** is still available for backward compatibility, consider using **LOGNORM.INV** from now on, because the former function may not be available in future versions of Excel
- if any argument is nonnumeric, **LOGNORM.INV** returns the **#VALUE!** error value
- if **probability** ≤ 0 or **probability** ≥ 1, **LOGNORM.INV** returns the **#NUM!** error value
- if **x** ≤ 0 or if **standard_deviation** ≤ 0, **LOGNORM.INV** returns the **#NUM!** error value.

Please see the example below:

	A	B	C
1	Data	Description	
2	0.269704931	Probability associated with the lognormal distribution	
3	2	Arithmetic mean of LN(x)	
4	1	Standard deviation of LN(x)	
5			
6			
7	Formula	Description	Result
8	=LOGNORM.INV(A2,A3,A4)	Inverse of the lognormal cumulative distribution function for the terms above	4.000000
9			

The A to Z of Excel Functions: LOGNORMDIST



The **LOGNORMDIST** function returns the Lognormal distribution of **x**, where **LN(x)** is Normally distributed with parameters **mean** and **standard_deviation**. The **LOGNORMDIST** function employs the following syntax to operate:

LOGNORMDIST(x, mean, standard_deviation)

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

- **x**: this is required and represents the value at which to evaluate the function
- **mean**: this is also required and denotes the mean of the natural logarithm of **x**, **LN(x)**
- **standard_deviation**: again required, this is the standard deviation of **LN(x)**.

It should be noted that:

- if any argument is nonnumeric, **LOGNORMDIST** returns the **#VALUE!** error value
- if **x** ≤ 0 or if **standard_deviation** ≤ 0, **LOGNORMDIST** returns the **#NUM!** error value
- the equation for the lognormal cumulative distribution function is:

$$\text{LOGNORMDIST}(x, \mu, \sigma) = \text{NORMSDIST}\left(\frac{\ln(x) - \mu}{\sigma}\right)$$

- This function has been replaced with a new function (**LOGNORM.DIST**) that may provide improved accuracy and whose names better reflect their usage. Although this function is still available for backward compatibility, you should consider using the new functions from now on, because this function may not be available in future versions of Excel.

Please see our final example for this month below:

	A	B	C
1	Data	Description	
2	4	Value for which you want the distribution	
3	2	Arithmetic mean of LN(x)	
4	1	Standard deviation of LN(x)	
5			
6			
7	Formula	Description	Result
8	=LOGNORMDIST(A2,A3,A4)	Cumulative lognormal distribution function for the terms above	0.269704931

More Excel Functions next month.

Beat the Boredom Suggested Solution

To recap, this month's challenge was to create multiple worksheets from a data list. The intention was to dynamically create the worksheets without manual intervention. Essentially, we wanted to generate multiple worksheets with the names from the data list shown below:

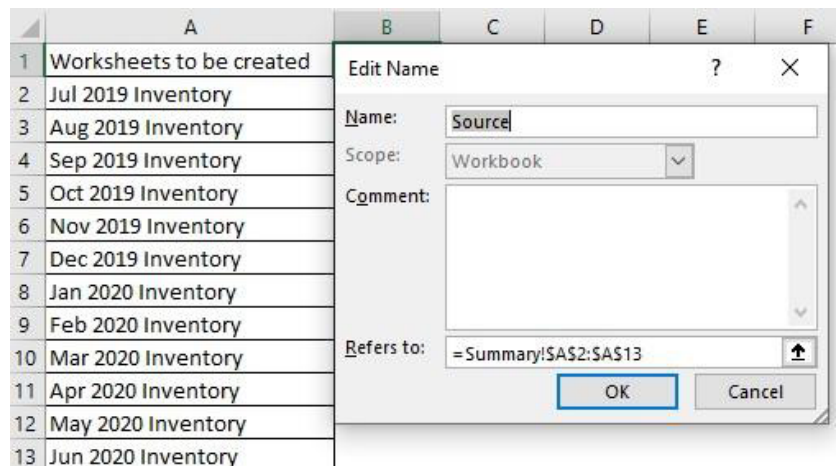
Worksheets to be created
Jul 2019 Inventory
Aug 2019 Inventory
Sep 2019 Inventory
Oct 2019 Inventory
Nov 2019 Inventory
Dec 2019 Inventory
Jan 2020 Inventory
Feb 2020 Inventory
Mar 2020 Inventory
Apr 2020 Inventory
May 2020 Inventory
Jun 2020 Inventory

to become:



Suggested Solution

One simple way here is to use VBA to look through the value of each cell in the data list and assign the value to each worksheet created after the 'Summary' worksheet (i.e. the sheet with the list). The first step is to set up a named range as shown below. Here, we will define the cell range **A2:A13** as the data list and assign a name tag **Source** to this range as shown below.



The next step is to declare the relevant variables.

```
Dim wksht As Worksheet
Dim rng As Range
Dim sName As String
```

Then, we set the variable **wksht** as the active worksheet and turn off the **ScreenUpdating** application. If the **ScreenUpdating** property were set to **False**, it will speed up the macro operation.

```
Set wksht =ActiveSheet
Application.ScreenUpdating = False
```

Next, we use a 'For Loop' to look up each value in the named range and assign the range value to the string variable **sName**. Then, we use an 'If' statement to determine the length of the variable **sName**. If the length is greater than zero, then we add a new worksheet after the last page and set the worksheet name equal to the value stored in variable **sName**, viz.

```
For Each rng In Range("Source")
sName = rng.Value
If Len(sName) > 0 Then
Worksheets.Add after:=Worksheets(Worksheets.Count)
ActiveSheet.Name = sName
End If
Next rng
```

Next, we go back to the **Summary** worksheet and turn on the **ScreenUpdating** application once more.

```
Worksheets("Summary").Select
Application.ScreenUpdating = True
```

Combing all the lines of code together, we get this:

```
Sub AddWorksheetsFromSelection()
    Dim wksht As Worksheet
    Dim rng As Range
    Dim sName As String
    Set wksht = ActiveSheet
    Application.ScreenUpdating = False
    For Each rng In Range("Source")
        sName = rng.Value
        If Len(sName) > 0 Then
            Worksheets.Add after:=Worksheets(Worksheets.Count)
            ActiveSheet.Name = sName
        End If
    Next rng
    Worksheets("Summary").Select
    Application.ScreenUpdating = True
End Sub
```

By using this method, we could create multiple worksheets from the named range without repetitive manual intervention.

Until 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	Date	Duration	Duration
Online (Australia)	Power Pivot, Power Query and Power BI	15 - 17 Aug 2023	09:00-17:00 AEST	(-1 day) 23:00-17:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	11 Sep 2023	09:00-17:00 AEST	(-1 day) 23:00-17:00 GMT	1 Day
Online (Australia)	Financial Modelling	12 - 13 Sep 2023	09:00-17:00 AEST	(-1 day) 23:00-17:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	22 - 24 Nov 2023	09:00-17:00 AEDT	(-1 day) 22:00-17:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	5 Dec 2023	09:00-17:00 AEDT	(-1 day) 22:00-17:00 GMT	1 Day
Online (Australia)	Financial Modelling	6 - 7 Dec 2023	09:00-17:00 AEDT	(-1 day) 22:00-17:00 GMT	2 Days

Key Strokes

Each newsletter, we'd like to introduce you to useful keystrokes you may or may not be aware of. We've started going through the alphabet actions. We earn a **D** this month...

Keystroke	What it does
CTRL + <u>D</u> ecimal	Rotates the active cell through the corners of the selection
<u>D</u> elete	Clear
CTRL + <u>D</u> elete	Clear
SHIFT + <u>D</u> elete	Cut
CTRL + <u>D</u> ivide (/)	Select array
<u>D</u> own Arrow	Move down one cell
ALT + <u>D</u> own Arrow	Open dropdown (AutoComplete, Filter or Validation)
CTRL + <u>D</u> own Arrow	Select the last cell in the area down
CTRL + ALT + <u>D</u> own Arrow	Intel chipset: invert screen (turn 180 degrees)
CTRL + SHIFT + <u>D</u> own Arrow	Extend selection down to last cell in area down
SHIFT + <u>D</u> own Arrow	Extend selection down one cell

There are c.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 www.sumproduct.com/training.

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

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