

Sum Product

NEWSLETTER #109 - December 2021

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

9 YEARS
108 MONTHS
3 2 8 7 DAYS
78,888 HOURS
4,733,280 MINUTES
283,996,800 SECONDS
of SumProduct

Our Newsletter turns Nine!! To celebrate our ninth anniversary, have we got lots of news for you this month!

In Excel updates, we report upon the improvements to Excel's add-ins and Data Types, we detail the new Feedback Portal, elaborate on the feature updates for Sheet View, Find All about Mac Excel and how Observable now integrates with Microsoft Excel Connector and O365 / OneDrive Cloud Files.

With a refresher on custom number formatting and another Beat the Boredom Challenge, there is also Charts & Dashboards, Visual Basics, Power Pivot Principles, Power Query Pointers, Power BI updates, we bring our function key series to an end in Keyboard Shortcuts and we go on for **HOURS** in this month's A to Z of Excel Functions.

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

Liam Bastick, Managing Director, SumProduct



Improvements to Excel Add-ins and Data Types

For many years, Microsoft Excel used to only allow two types of data: text and numbers. But that's changing: Microsoft is opening up Excel to provide you with the freedom to create your own custom data types that can contain images, arrays and more.

A new JavaScript API was made available in Preview inside Excel in late November, which will allow you to create your own add-ins or extend previously existing ones to utilise the power of Data Types, resulting in a more integrated and next-generation experience within Excel. You can share these Data Types across your entire organisation and create add-ins or solutions which can connect data types to your own service or data.

In essence, Excel developers will be able to create new add-ins or update existing ones to use the JavaScript API.

This will open Excel up to far more custom data types, including content cards, images, matrices, arrays, and formatted number values. Excel has long had support for macros and add-ins, but Microsoft's new APIs should make this a lot more efficient and easier for developers to implement.

The following is the complete list of types supported in the initial release:

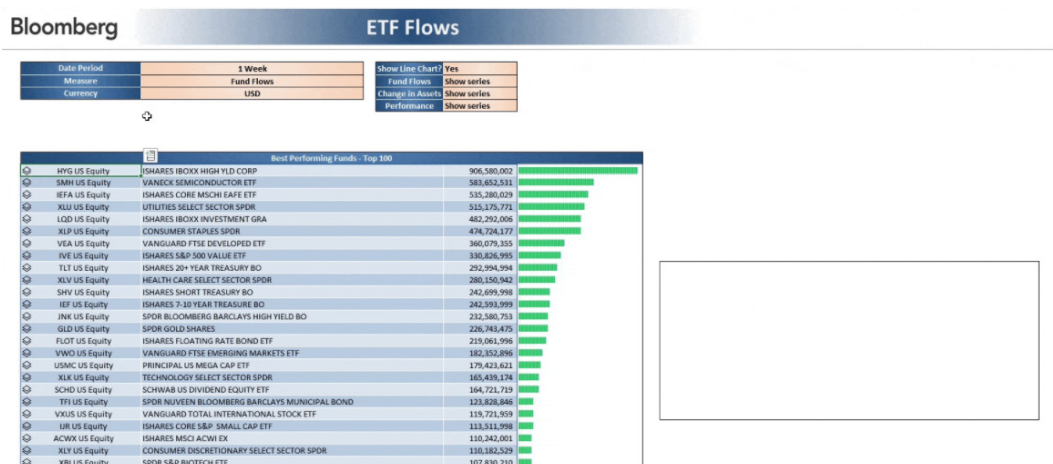
- entity values
- formatted number values
- web images
- improved errors
- arrays (only supported as a property of an entity and cannot be a standalone value currently).

With these value types, you are free to use them as both inputs and outputs for your experiences, and you can also arbitrarily combine them.

For example, you can create an entity value that contains properties of all the types, such as a web image for richer context about the entity, formatted values for things like currencies or dates, arrays of types for lists of data, and even additional entities through nesting.

This allows you to take your flat data and package it up into a logical value, as shown in the following example.

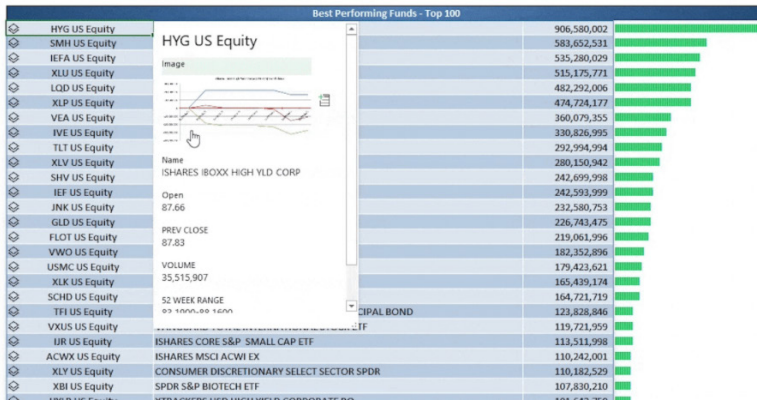
As an illustration, consider the following use of the API by Bloomberg:



By clicking on the icon next to 'HYG US Equity', you can inspect the Rich Data Type's card, viz.

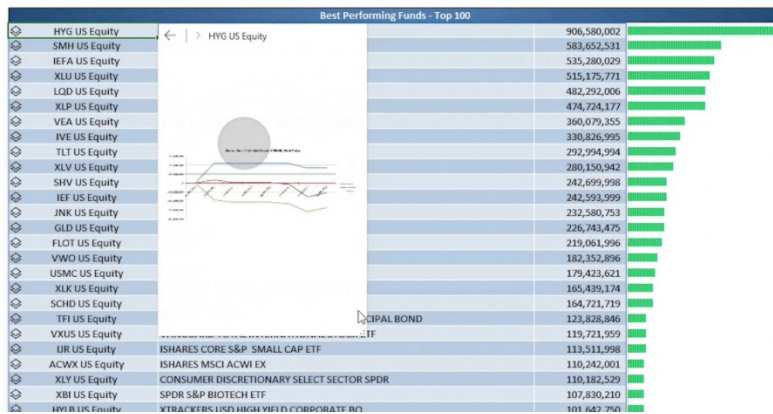


Date Period	1 Week	Show Line Chart?	Yes
Measure	Fund Flows	Fund Flows	Show series
Currency	USD	Change in Assets	Show series
		Performance	Show series



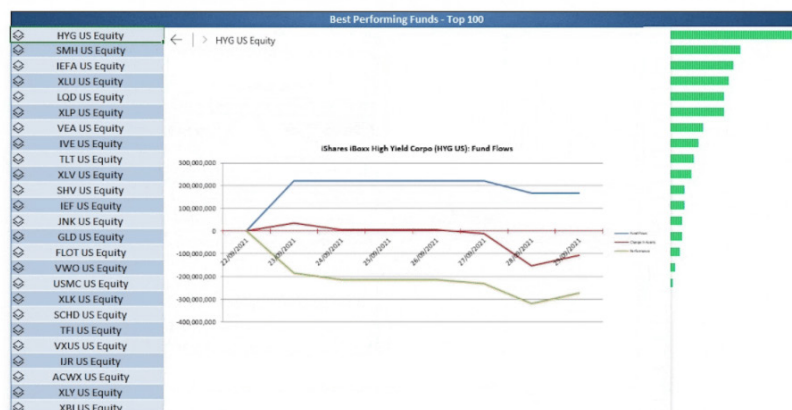
If you click on the image presented, you can see the Fund Flows diagram:

Date Period	1 Week	Show Line Chart?	Yes
Measure	Fund Flows	Fund Flows	Show series
Currency	USD	Change in Assets	Show series
		Performance	Show series

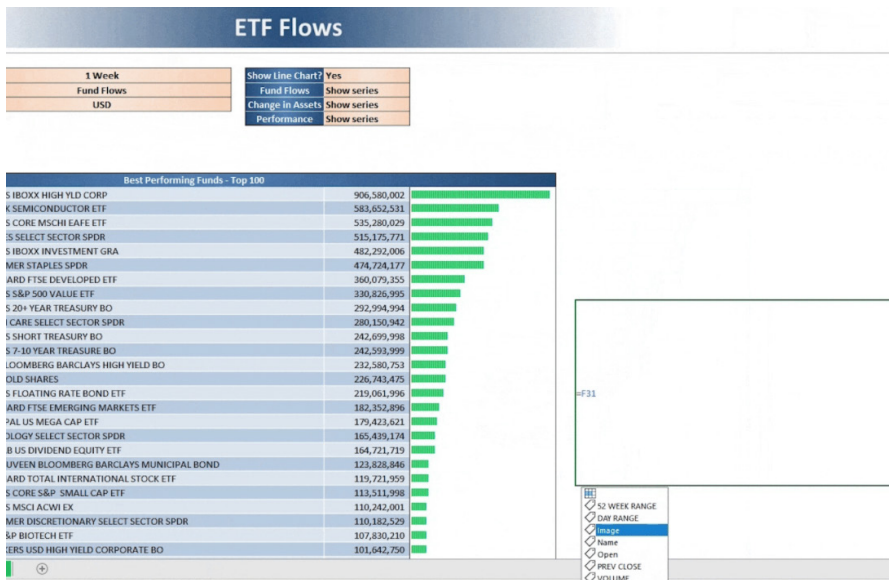


You can make it bigger

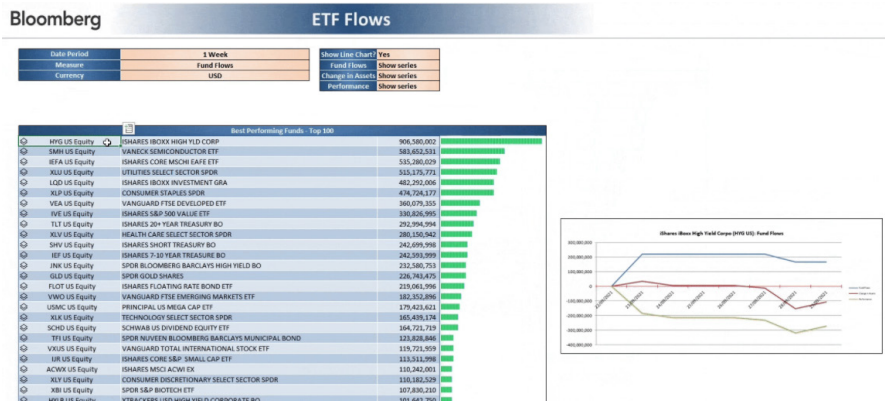
Date Period	1 Week	Show Line Chart?	Yes
Measure	Fund Flows	Fund Flows	Show series
Currency	USD	Change in Assets	Show series
		Performance	Show series



or else reference it formulaically (see the boxed area on the right, referring to cell F31):



Et voila!



The first API being introduced, `Range.valuesAsJson`, allows you to both read and write data types, which works similarly to the `Range.values` API already in circulation. The major difference is that this new API can return augmented information about basic types (text, numbers, errors) as well as information about the new data types we have introduced. One other notable difference is the introduction of a schema that conforms to the types that are available to Excel.

Microsoft's focus has been on exposing the structures so that you can bring your data to Excel using the schema and leverage many of the built-in experiences today. These structures are all driven by the schema. Here are some quick examples of JSON that adheres to the schema and creates a few types. Be sure to take note of the type field, which tells Excel what type a given value is:

<pre>const myImg = { type: "WebImage", address: "https://bit.ly/2YG0wtw" };</pre>	
<pre>const myDate = { type: Excel.CellValueType.formattedNumber, basicValue: 32889.0, numberFormat: "m/d/yyyy" };</pre>	
<pre>const myEntity = { type: Excel.CellValueType.entity, text: "A llama", properties: { image: myImg, "start date": myDate, "quote": { type: Excel.CellValueType.string, basicValue: "I love Excel llambdas" } } };</pre>	

Upon release, to access these new APIs, you will need:

- a valid Insiders: Beta build of Excel
- Windows version greater than or equal to 16.0.14626.10000 / Mac version greater than or equal to 16.55.21102600
- the latest preview version of the js APIs.

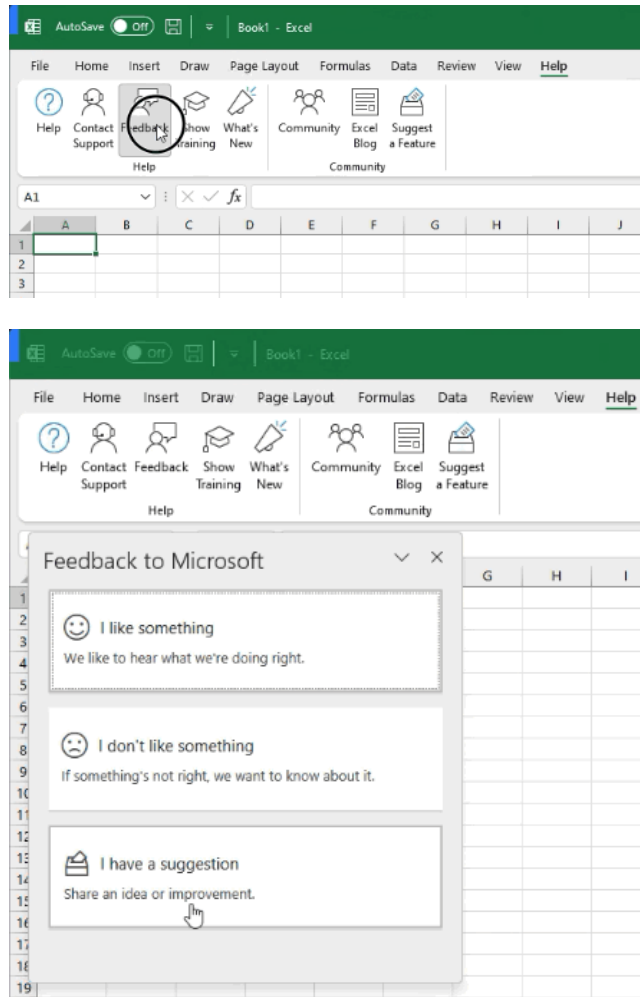
New Microsoft Feedback Portal

To replace Excel UserVoice, the software giant has introduced their new Microsoft Feedback portal, where you may submit suggestions about Excel and other Microsoft products, vote and comment on suggestions made by others, and see feedback submitted.

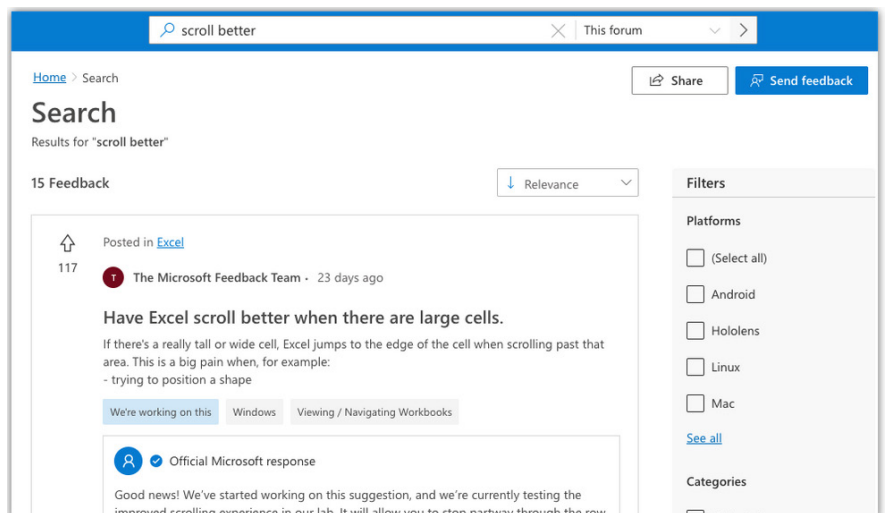
Microsoft states they have "...brought the top ideas over to the Excel forum on the new Feedback portal...", but there do appear to have been

a lot of suggestions go walkabout. The idea is that if you were following the ideas submitted on UserVoice, you should see lots of similar ideas and discussions on the new site.

Getting started with the Feedback portal is fairly straightforward. Just press the Feedback button in Excel, and choose 'I have a suggestion'.

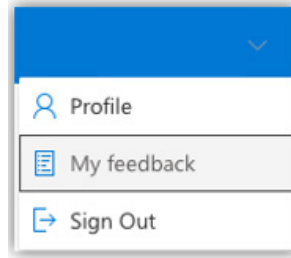


You'll be taken to the Excel forum on the Feedback portal, where you may submit new ideas for the community, or participate by voting for existing suggestions, and adding your comments to the conversation.



Users may access the Feedback portal whether they're on a desktop or mobile device, irrespective of a particular device or form factor. Further, you may also continue sending feedback from within apps, where it is available.

Do note the section called 'My Feedback' (click your profile button) where you may view feedback you've submitted publicly on the site, or privately from within the applications.



However, if you don't get taken to the new Feedback site, it may be your organisation may be using a different configuration for how you should provide feedback to Microsoft.

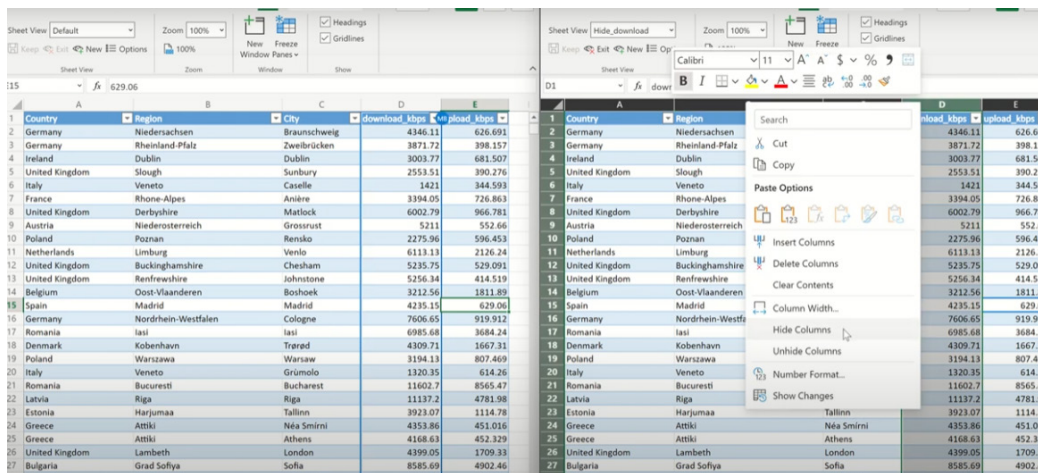
The Preview of Feedback is available for Excel and many other Microsoft apps and services.

Feature Updates for Sheet View

If you have ever collaborated with someone else in a workbook and suddenly notice that the data table you're looking at has changed drastically, this often happens because someone has filtered or sorted the table while you were looking at the data. Microsoft accepts that while people want to work together in a file, it doesn't necessarily mean everyone has the same task at hand. To provide users with the freedom to collaborate as they want, they have previously developed Sheet View. With Sheet View, you can create customised views in an Excel workbook

to freely sort and filter without being impacted by others. That has been around for a while, but now there have been some recent improvements made to Sheet View.

One of the top requests for Excel for the Web was to support hiding rows and columns in a Sheet View, as well as expanding and collapsing grouped rows and columns. These actions are now supported while working in a Sheet View across all platforms.



Furthermore, with people working from anywhere and on different devices, Sheet View is now also supported on iOS, iPad and Android.



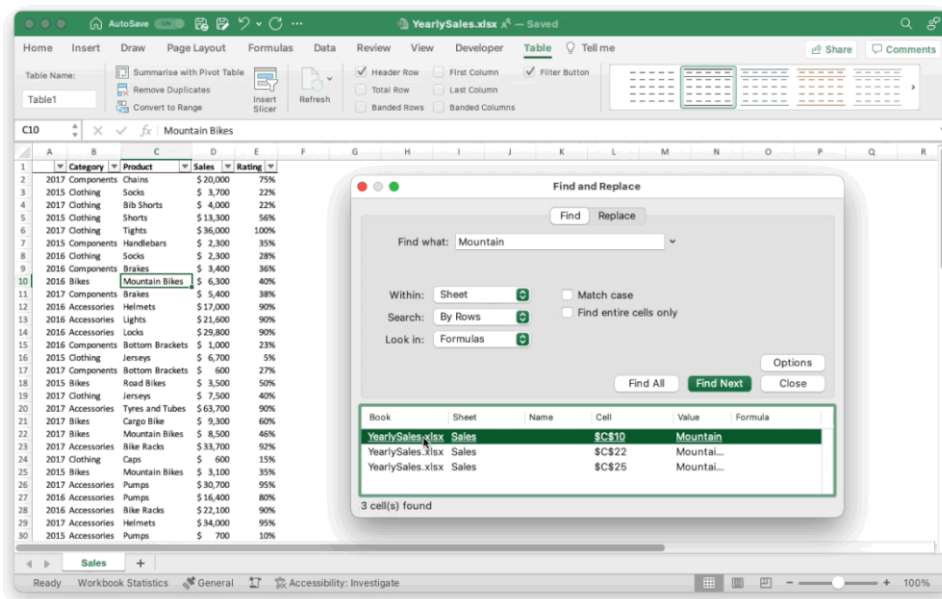
Hiding rows and columns in Sheet View is supported on Excel desktop builds greater than or equal to 16.0.14005.10000, July semi-annual build, and June MEC build. Sheet View iOS mobile and Android are available in 100% production as at the time of writing.

Find All in Mac Excel

For a while, the Microsoft Excel team has stated they have been trying to improve Excel for Mac, and their latest endeavours sees some improvements to the Find and Replace feature: you may now edit cells without having to close the Find dialog first. But that's not all – they've decided to go and spoil Mac users completely and provide them with the luxurious Find All too!

To use Find All in Excel for Mac:

- open the 'Find and Replace' dialog, using **CMD + F** or **Home -> Find & Select -> Find**
- enter your search term and click the 'Find All' button
- you will notice that the dialog expands to show a list of all the cells that contain the search term, and the total number of cells in which it appears.
- click any item in the list to select the corresponding cell in your worksheet.

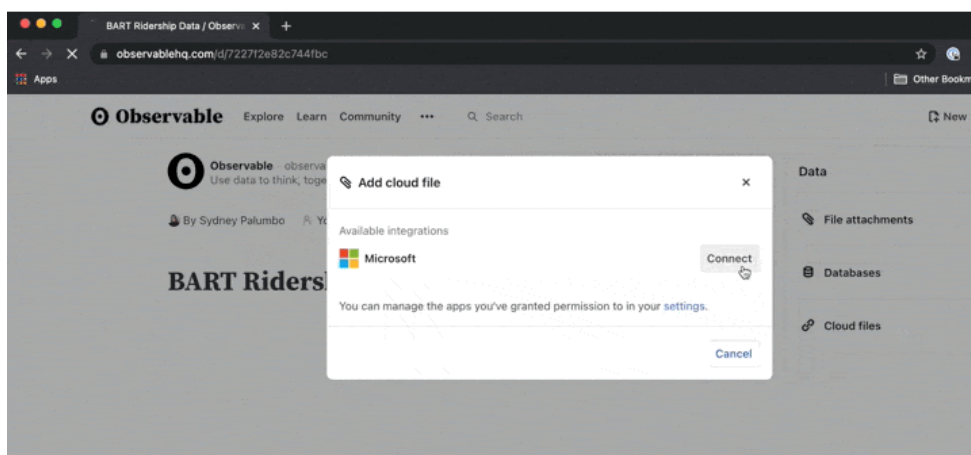


This improvement is currently being made available to Office Insiders only.

Observable now integrates with Microsoft Excel Connector and O365 / OneDrive Cloud Files

Observable's new integrations with Excel, O365 and OneDrive allow users to bring data from Microsoft sources to the Observable platform, enabling real time data collaboration and analysis.

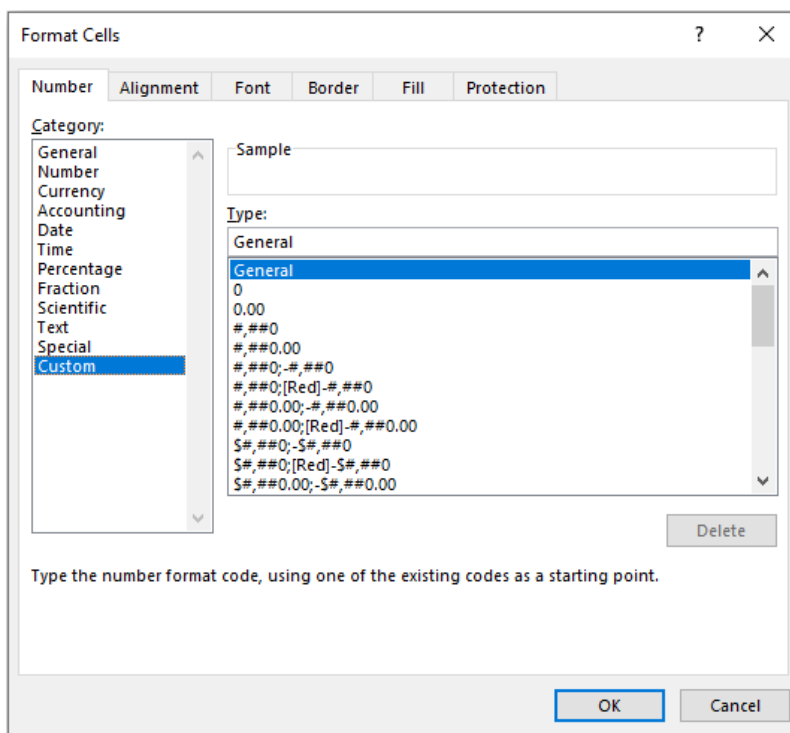
The Excel connector supports a local Excel file to be uploaded and parsed in Observable. Users may call `.xlsx()` on a file attachment and access individual sheets of the Excel file, as well as upload zip files or multiple spreadsheets at a time by using `FileAttachment.zip()`.



Additionally, Observable now supports accessing data through cloud files. This enables users to reference live files and bring in data from O365 and OneDrive. The data from these files may be refreshed when a spreadsheet is updated.

Custom Number Formatting

Talking of Data Types, we have had a few questions on custom number formatting recently, so we thought it was time to give it another mention in the newsletter. Given that one of the primary purposes of a spreadsheet is to present numerical data, it is important how numerical data is presented. Cells may be individually formatted, using **CTRL + 1** (all versions of Excel):



Formatting only changes the appearance, not the underlying value, of a cell. For example, if cells **A1** and **B1** had the number '1.4' typed in but were formatted to zero decimal places, then if cell **C1 = A1 + B1**, you would truly have $1 + 1 = 3$ (well, $1.4 + 1.4 = 2.8$ anyway).

This should not be confused with '(Set) precision as displayed' (Excel 2007 and later: **ALT + T + O -> Advanced -> When calculating this workbook -> Set precision as displayed**; earlier Excel versions: **Tools -> Options -> Calculation -> Workbook options -> Precision as displayed**). Selecting this option and clicking 'OK' will permanently change stored values in cells to whatever format has been selected, including the number of decimal places (e.g. 15.75 formatted to one decimal placed would become precisely 15.8).

Excel has many built-in number formats that are fairly easy to

understand, e.g. Currency, Date, Percentage. The default format is 'General' where Excel will endeavour to provide the most appropriate format for the contents. For example, typing '3 3/4' into a cell will result in Excel selecting a mixed format.

But what do you do if you can't find an appropriate format?

Selecting the 'Custom' category activates the 'Type' input box and allows between 200 and 250 custom number formats in a particular workbook, depending upon the language version of Excel that has been installed.

The 'Type' input box allows up to four aspects of formatting to be specified in a cell. These aspects are referred to as sections and are separated by a semi-colon (;). To ascertain what is contained in each section depends on the total number of sections used, viz.

No. of Sections	Section Details (assuming no conditions)
1 (min)	All numerical values
2	Non-negative Numbers; Negative Numbers
3	Positive Numbers; Negative Numbers; Zero Values
4 (max)	Positive Numbers; Negative Numbers; Zero Values; Text

To the uninitiated, coding custom number formats may appear incomprehensible. However, understanding the following tables from Microsoft soon puts things into perspective.

Number Code	Description
General	General number format
0	Digit placeholder (if no number, a '0' will be used to 'pad')
#	Digit placeholder (does not display extra zeros)
?	Digit placeholder (leaves space for extra zeros, but does not display them)
. (decimal point / full stop)	Decimal point!
%	Percentage displayed
, (comma)	Thousands separator
/	Used to delineate numerator from denominator in Fraction category
E+ e+ E- e-	Scientific notation

Text Code	Description
\$ - + / () : space	These characters are displayed in the number.
"text"	For other characters, in order to ensure Excel does not misinterpret them, it is best to use enclose the character(s) in quotation marks...
\character	...or precede it with a backslash
*	Repeats the next character in the format to fill the column width. Only one asterisk per section of a format is allowed
_character	Skips the width of the next character. In particular, this syntax is often used with the closing parenthesis,), in a positive number format (when the negative format includes brackets). This allows the values to line up at the decimal point
@	Text placeholder

Date Code	Description
m	Month as a number without leading zeros (1 to 12)
mm	Month as a number with leading zeros (01 to 12)
mmm	Month as an abbreviation (Jan – Dec)
mmmm	Unabbreviated month (January – December)
mmmmm	First letter of month (J, F, M, A, M, J, J, A, S, O, N, D)
d	Day without leading zeros (1 to 31)
dd	Day with leading zeros (01 to 31)
ddd	Weekday as an abbreviation (Sun – Sat)
dddd	Unabbreviated weekday (Sunday – Saturday)
y or yy	Year as a two digit number (e.g. 09, 97)
yyy or yyyy	Year as a four digit number (e.g. 2009, 1997)

Time Code	Description
h	Hours as a number without leading zero (0 to 23)
hh	Hours as a number with leading zero (00 to 23)
m	Minutes as a number without leading zero (0 to 59)
mm	Minutes as a number with leading zero (00 to 59)
s	Seconds as a number without leading zero (0 to 59)
ss	Seconds as a number with leading zero (00 to 59)
[h]	With times only, will increment hours to 24 and beyond
[m]	With times only, will increment minutes to 60 and beyond
[s]	With times only, will increment seconds to 60 and beyond
AM/PM am/pm	Time based on the 12-hour clock [24-hour clock is the default]

Miscellaneous Code	Description
[Black], [Blue], [Cyan], [Green], [Magenta], [Red], [White], [Yellow]	Displays the characters in the specified colours
[Color <i>n</i>]	Displays the characters in a specified colour, where <i>n</i> is a value from 1 to 56, and refers to the <i>n</i> th colour in the colour palette
;	Delineates a section
[Condition Value]	<i>Condition</i> may be any one of the comparison operators, <, >, =, <=, >=, <> and <i>Value</i> may be any number. A number format may contain up to two conditions

For more on number formatting, illustrative examples and to download a free sample Excel workbook, please go to <http://www.sumproduct.com/thought/number-formatting>.

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...

You may have worked with PivotTables all of the time and believe you know them inside out. However, there may be a small filtering option that you may not know of. Do you know how to filter PivotTables based on values in **all** columns?

	A	B	C	D	E	F
1						
2						
3		Sum of Total Invoice Column Labels ▾				
4		Row Labels	Accessories ▾	Bikes ▾	Clothing ▾	Grand Total ▾
5		Australia	\$153,253.45	\$9,781,515.48	\$77,637.33	\$10,012,406.26
6		Canada	\$114,232.75	\$2,012,539.19	\$58,746.97	\$2,185,518.90
7		France	\$70,064.64	\$2,821,701.23	\$29,873.95	\$2,921,639.83
8		Germany	\$68,767.16	\$3,103,408.43	\$26,039.80	\$3,198,215.39
9		United Kingdom	\$84,676.38	\$3,627,541.23	\$35,624.71	\$3,747,842.32
10		United States	\$283,346.93	\$9,944,845.02	\$147,526.40	\$10,375,718.35
11		Grand Total	\$774,341.31	\$31,291,550.59	\$375,449.16	\$32,441,341.06
12						

Applying filters on a table is relatively easy but are you able to achieve the same thing in a PivotTable?

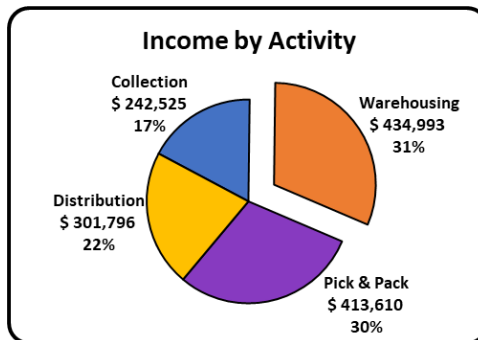
Sum of Total Invoice	Column Labels		
Row Labels	Accessories	Bikes	Clothing
Canada			\$58,746.97
France			\$29,873.95
Germany			\$26,039.80
			\$2,185,518.90
			\$2,921,639.83
			\$3,198,215.39

This month's challenge is to be able to filter on **all** columns in a PivotTable in Excel with no VBA, no megaformulae, and not in Power BI, but in Excel. Sound 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 look at Pie of Pie charts.

Apart from Pie Charts, Excel also offers 'Pie of Pie' and 'Bar of Pie' chart variations. Below is the Pie Chart where Warehousing stands out as the largest wedge, which was created in last month's newsletter:



Imagine that I want to further breakdown the Warehousing income by customer in the one chart. Setting up the data correctly is pivotal. To start, instead of having total Warehousing income, I need to replace it

with the breakdown by customer, and this data *must* be at the bottom of the data table:

Category	2017/18 Income
Collection	\$ 242,525
Pick & Pack	\$ 413,610
Distribution	\$ 301,796
BizSupplies	\$ 128,194
Dynamic Attire	\$ 81,318
Harmonic Sonics	\$ 66,315
Plumb'n'Stuff	\$ 91,765
Other Clients	\$ 67,401

These rows contain the amounts of Warehousing income by customer

These rows contain the total income for the income types other than Warehousing.

However, the categories are now out of sequence, as I have Collection, then Pick & Pack, Distribution and finally Warehousing, where Warehousing should be between Collection and Pick & Pack. Note that the data is being plotted in a circle. It doesn't matter which item is at the top or bottom so long as the cyclical order is maintained. Therefore, the top section of the table needs to be reorganised so that Collection income

is immediately above the Warehousing income lines and then where Pick & Pack Distribution would normally be below Warehousing, they will now be at the top of the data table so that I keep the Warehousing income lines at the bottom for creating this Pie of Pie chart.

The final data table would look like the one below:

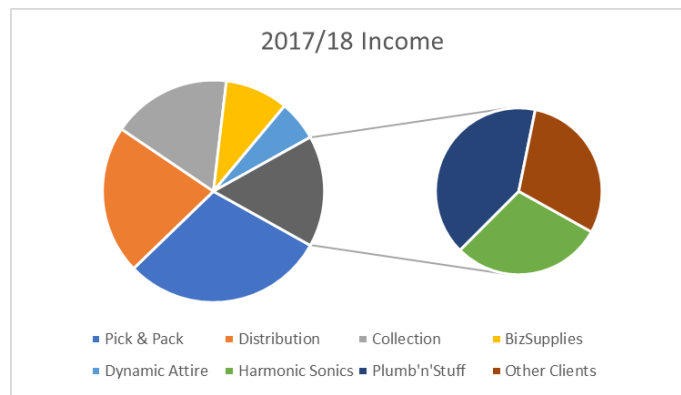
Category	2017/18 Income
Pick & Pack	\$ 413,610
Distribution	\$ 301,796
Collection	\$ 242,525
BizSupplies	\$ 128,194
Dynamic Attire	\$ 81,318
Harmonic Sonics	\$ 66,315
Plumb'n'Stuff	\$ 91,765
Other Clients	\$ 67,401

These rows contain the amounts of Warehousing income by customer

3rd Income Type
4th Income Type
1st Income Type

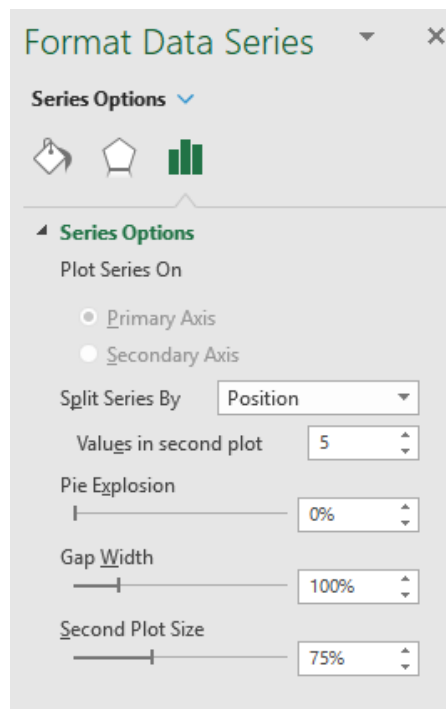
2nd Income Type
These rows must be at the bottom of the data table

Having the data table ready, I highlight it and click Insert, choose the 'Pie Chart' icon. This time, I choose the 'Pie of Pie' variation and the following chart should appear:

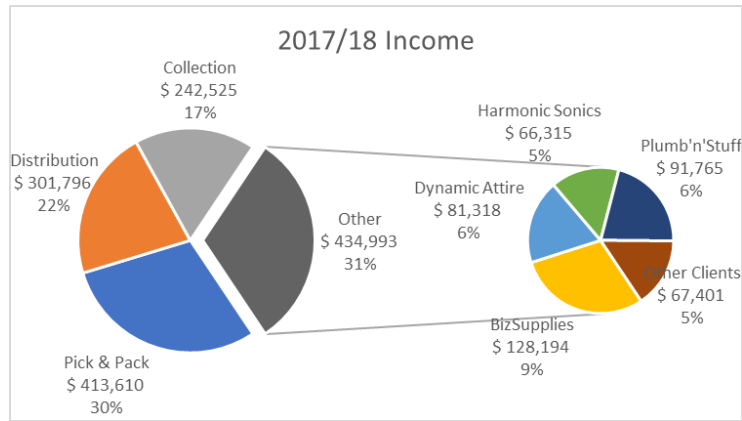


This chart is initially confusing until I match the legend to the segments and realise that Excel has only taken the bottom three rows of the data table and plotted these in the smaller Pie Chart, so I need to configure the chart.

I right-click on the chart and choose 'Format Data Series', then set the 'Values in the second plot' to be five, to match the number of rows relating to Warehousing income in my source data.

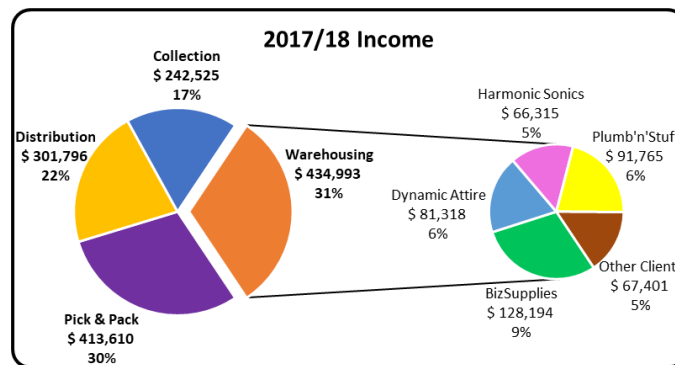


Let's also add data labels and format them the same as I did for the first Pie Chart at the beginning, remove the Legend and make Warehousing "explode" slightly.



Here, the Warehousing segment is labelled Other, as Excel does not have any wording to apply to this wedge from my data table. To change the label, click on the any of the data labels. All of them will be selected. Then, clicking a second time on the Other data label will release the

other three labels. I click the third time on the Other label and now I can modify the text within this label. I also move the 'Other Clients' label in the smaller Pie Chart so that it does not overlap the wedge. Once this is done, I do some further formatting as usual and here is my 'double pie':



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 continue looking at DataBodyRange – this time, with more than a nod to Star Trek...

Not wanting to Troi too hard, let's continue with Data(BodyRange)! (Warning: it's going to be bad this month – Ed.)

First, let's examine consider our table:



Oops! Not that one...

1	Name	Media	Release Year
2	The Original Series	Television	1966
3	The Animated Series	Television	1973
4	The Next Generation	Television	1987
5	Deep Space Nine	Television	1993
6	Voyager	Television	1995
7	Enterprise	Television	2001
8	Discovery	Television	2017
9	Star Trek: The Motion Picture	Film	1979
10	Star Trek II: The Wrath of Khan	Film	1982
11	Star Trek III: The Search for Spock	Film	1984
12	Star Trek IV: The Voyage Home	Film	1986
13	Star Trek V: The Final Frontier	Film	1989
14	Star Trek VI: The Undiscovered Country	Film	1991
15	Star Trek Generations	Film	1994
16	Star Trek: First Contact	Film	1996
17	Star Trek: Insurrection	Film	1998
18	Star Trek: Nemesis	Film	2002
19	Star Trek	Film	2009
20	Star Trek Into Darkness	Film	2013
21	Star Trek Beyond	Film	2016

Let's filter it down to the bare-Bones to only media released before 1990:

1	Name	Media	Release Year
2	The Original Series	Television	1966
3	The Animated Series	Television	1973
4	The Next Generation	Television	1987
9	Star Trek: The Motion Picture	Film	1979
10	Star Trek II: The Wrath of Khan	Film	1982
11	Star Trek III: The Search for Spock	Film	1984
12	Star Trek IV: The Voyage Home	Film	1986
13	Star Trek V: The Final Frontier	Film	1989

Back to VBA, if the information in the table is copy pasted, what would the expected result be? Let's take a Quark at it.

```

(General)
Option Explicit

Sub DataBodyRange2 ()

    Dim myTable As ListObject
    Set myTable = Range("StarTrekTable").ListObject

    myTable.DataBodyRange.Copy

    Range("A25").PasteSpecial xlPasteValues

End Sub

```


After running the code, the result is as follows:

Name	Media	Release Year
The Original Series	Television	1966
The Animated Series	Television	1973
The Next Generation	Television	1987
Star Trek: The Motion Picture	Film	1979
Star Trek II: The Wrath of Khan	Film	1982
Star Trek III: The Search for Spock	Film	1984
Star Trek IV: The Voyage Home	Film	1986
Star Trek V: The Final Frontier	Film	1989
The Original Series	Television	1966
The Animated Series	Television	1973
The Next Generation	Television	1987
Deep Space Nine	Television	1993
Voyager	Television	1995
Enterprise	Television	2001
Discovery	Television	2017
Star Trek: The Motion Picture	Film	1979
Star Trek II: The Wrath of Khan	Film	1982
Star Trek III: The Search for Spock	Film	1984
Star Trek IV: The Voyage Home	Film	1986
Star Trek V: The Final Frontier	Film	1989
Star Trek VI: The Undiscovered Country	Film	1991
Star Trek Generations	Film	1994
Star Trek: First Contact	Film	1996
Star Trek: Insurrection	Film	1998
Star Trek: Nemesis	Film	2002
Star Trek	Film	2009
Star Trek Into Darkness	Film	2013
Star Trek Beyond	Film	2016

Notice how it has copied ALL the data. You wouldn't expect that would you?! Computers make excellent and efficient servants, but I have no wish to serve under them.

Let's Forge ahead and get just the filtered Data. In order to do that, let's adjust the code slightly to use the .SpecialCells method of a Range. This allows selection of a particular type of cells. Given that only visible cells from the filter is required, the xlCellTypeVisible is the property needed.

```

(General)
Option Explicit

Sub DataBodyRange2 ()

    Dim myTable As ListObject
    Set myTable = Range("StarTrekTable").ListObject

    myTable.DataBodyRange.SpecialCells(xlCellTypeVisible).Copy

    Range("A25").PasteSpecial xlPasteValues

End Sub

```

Et voila!

1	Name	Media	Release Year
2	The Original Series	Television	1966
3	The Animated Series	Television	1973
4	The Next Generation	Television	1987
9	Star Trek: The Motion Picture	Film	1979
10	Star Trek II: The Wrath of Khan	Film	1982
11	Star Trek III: The Search for Spock	Film	1984
12	Star Trek IV: The Voyage Home	Film	1986
13	Star Trek V: The Final Frontier	Film	1989
22			
23			
24			
25	The Original Series	Television	1966
26	The Animated Series	Television	1973
27	The Next Generation	Television	1987
28	Star Trek: The Motion Picture	Film	1979
29	Star Trek II: The Wrath of Khan	Film	1982
30	Star Trek III: The Search for Spock	Film	1984
31	Star Trek IV: The Voyage Home	Film	1986
32	Star Trek V: The Final Frontier	Film	1989

For the SpecialCells methods other properties may be used as well, for example:

- xlCellTypeFormulas - cells containing only formulae
- xlCellTypeBlanks - empty cells
- xlCellTypeLastCell - last cell in the range.

More next time.

Power Pivot Principles

We continue our series on the Excel COM add-in, Power Pivot. This month, we discuss some tips and best practices you should keep in mind when working with calculated columns.

When creating new calculated columns, they are automatically added to the far right of other columns in a table inside the Power Pivot window, e.g.

	ProductSubCategoryAlternateKey	EnglishProductSubCategoryName	ProductCategoryK	Standard Cost	Sales Amount	Calculated Column 1
1	1	Sun Solution - light	1	35.5	79.95	44.45
2	2	Sun Solution - medium	1	35.5	79.95	44.45
3	3	Sun Solution - dark	1	35.5	79.95	44.45
4	4	Violet Base Solution - light	1	39.95	89.95	50

Their positions are not permanent though; we may rearrange the calculated columns after we have created them, viz.

	Product	ProductSubCategoryAlternateKey	EnglishProductSubCategoryName	ProductCategoryK	Calculated Column 1	Standard Cost	Sales Amount
1	1	1	Sun Solution - light	1	44.45	35.5	
2	2	2	Sun Solution - medium	1	44.45	35.5	
3	3	3	Sun Solution - dark	1	44.45	35.5	
4	4	4	Violet Base Solution - light	1	50	39.95	

New Calculated Columns are assigned the default name of 'Calculated Column 1', 'Calculated Column 2', etc. Therefore, we should always give calculated columns a unique name within each table. Otherwise, using them in measures or other calculated columns may lead to calculation issues.

You should always give a unique name to a calculated column immediately after creating it. If we rename the calculated column after using it in other calculated columns or measures we may also have to update any formulae that relied on the initial calculated column.

Calculated columns can possess the same name as measures, however doing

so may result in potential calculation errors down the road. We strongly recommend avoiding assigning the same name to a calculated column and a measure.

Furthermore, when referring to calculated columns in other calculated columns or measures we should always use the 'fully qualified column reference'. To be clear, a 'fully qualified column reference' includes the table name followed by the column name in square brackets. For example, the fully qualified column reference for the 'Standard Cost' column in the 'Product SubCategory' table is:

='Product SubCategory'[Standard Cost]

This is so that we can clearly distinguish between calculated columns, columns (field names) and measures in our formulae.

More *Power Pivot Principles* next month.

Power Query Pointers

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

This month, let's look at a simple scenario where we may use a parameter in Power Query which is populated from a cell in my Excel workbook. Below, I have a spreadsheet with charge data pertaining to items in my fictitious inventory:

Charge_Line_Number	Amount	Discount	Item_Key	Order_Detail_Key	Description
1	50		1 2-1		Side Connecting Porch
3	50		1 2-2		Side Connecting Porch
4	50		1 2-3		Side Connecting Porch
5	50	0	1 2-31		Side Connecting Porch
6	50	0	1 2-32		Side Connecting Porch
7	50	0	1 2-33		Side Connecting Porch
8	120		2 1-2		3 x3 metre marquee
9	120		2 1-15		3 x3 metre marquee
10	120		2 1-17		3 x3 metre marquee
11	120		2 1-26		3 x3 metre marquee
12	120	0	2 1-32		3 x3 metre marquee
13	120	0	2 1-45		3 x3 metre marquee
14	120	0	2 1-47		3 x3 metre marquee
15	120	0	2 1-56		3 x3 metre marquee
16	150		3 1-4		4 x 4 metre marquee
17	150		3 1-11		4 x 4 metre marquee
18	150		3 1-12		4 x 4 metre marquee
19	150		3 1-20		4 x 4 metre marquee
20	150		3 1-25		4 x 4 metre marquee
21	150	0	3 1-34		4 x 4 metre marquee
22	150	0	3 1-41		4 x 4 metre marquee
23	150	0	3 1-42		4 x 4 metre marquee
24	150	0	3 1-50		4 x 4 metre marquee

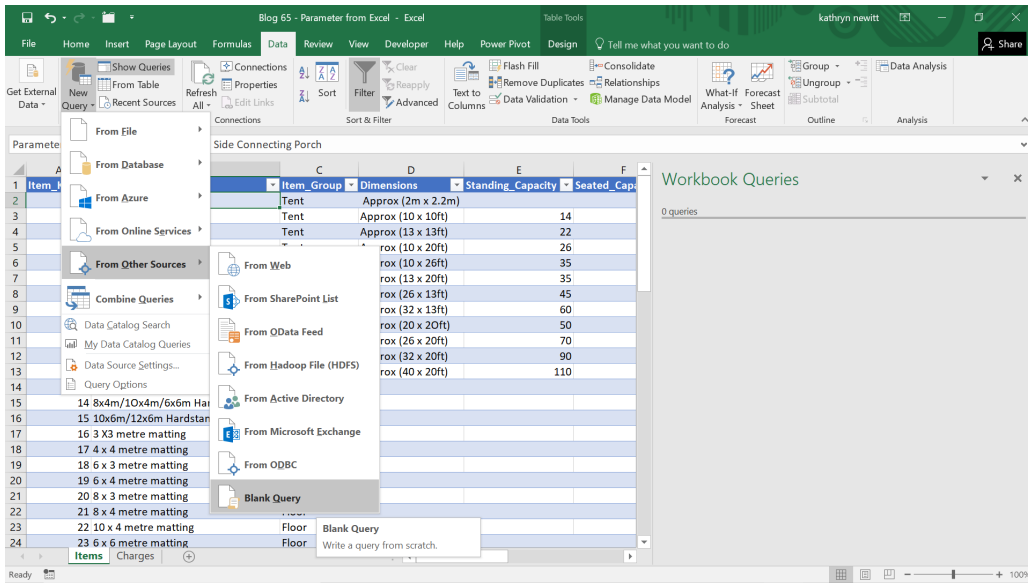
The plan is to create a query from 'Charges' data, and pass a parameter from the 'Items' sheet.

Item_Key	Item_Name	Item_Group	Dimensions	Standing_Capacity	Seated_Cap.
1	Side Connecting Porch	Tent	Approx (2m x 2.2m)		
2	3 x3 metre marquee	Tent	Approx (10 x 10ft)	14	
4	3 x 4 metre marquee	Tent	Approx (13 x 13ft)	22	
5	4 x 3 metre marquee	Tent	Approx (10 x 20ft)	26	
6	5 x 3 metre marquee	Tent	Approx (10 x 26ft)	35	
7	6 x 4 metre marquee	Tent	Approx (13 x 20ft)	35	
8	7 x 4 metre marquee	Tent	Approx (26 x 13ft)	45	
9	8 10 x 4 metre marquee	Tent	Approx (32 x 13ft)	60	
10	9 6 x 6 metre marquee	Tent	Approx (20 x 20ft)	50	
11	10 8 x 6 metre marquee	Tent	Approx (26 x 20ft)	70	
12	11 10 x 6 metre marquee	Tent	Approx (32 x 20ft)	90	
13	12 12 x 6 metre marquee	Tent	Approx (40 x 20ft)	110	
14	13 4x4m/6x3m/6x4m Hardstanding kit	Floor			
15	14 8x4m/10x4m/6x6m Hardstanding kit	Floor			
16	15 10x6m/12x6m Hardstanding kit	Floor			
17	16 3 X3 metre matting	Floor			
18	17 4 x 4 metre matting	Floor			
19	18 6 x 3 metre matting	Floor			
20	19 6 x 4 metre matting	Floor			
21	20 8 x 3 metre matting	Floor			
22	21 8 x 4 metre matting	Floor			
23	22 10 x 4 metre matting	Floor			
24	23 6 x 6 metre matting	Floor			

No one is claiming that this is the most efficient approach for this particular scenario, but it will serve to show the method!

We'll create the parameter first, so that it is ready to add to our main 'Charges' query. Since the cell is already in a table of items data, trying to create a new query from the 'Data' tab by choosing 'From Table' in the

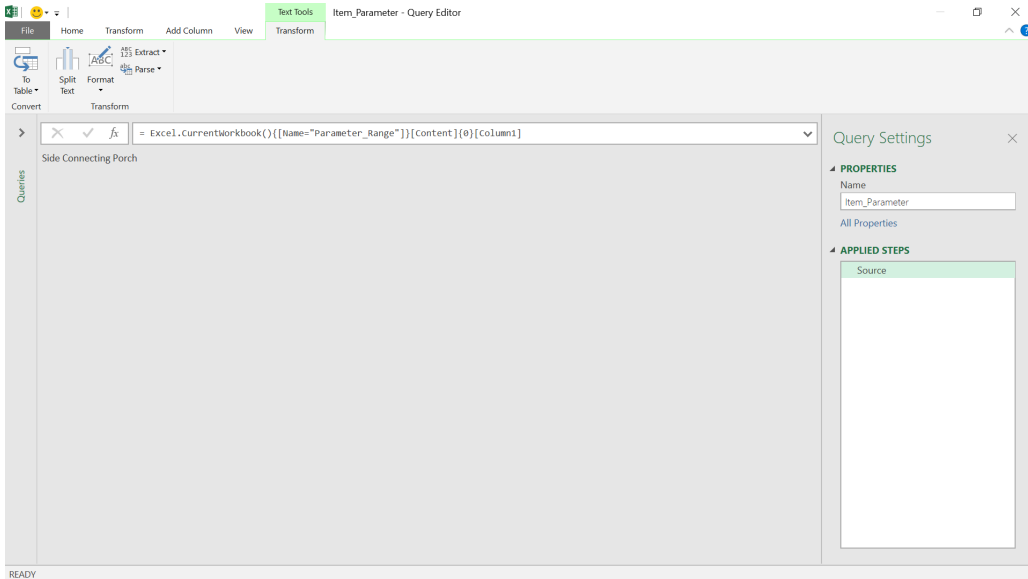
'Get and Transform' section will assume we want the whole table. We may circumvent this issue by using the method previously used in [Power Query: Returning to Referencing Ranges](#). If we give our prospective parameter a range name, then we may locate it easily in the workbook. Let's name the cell 'Parameter_Range' and create a new blank workbook from the 'Other sources' option of the 'New Query' dropdown, viz.



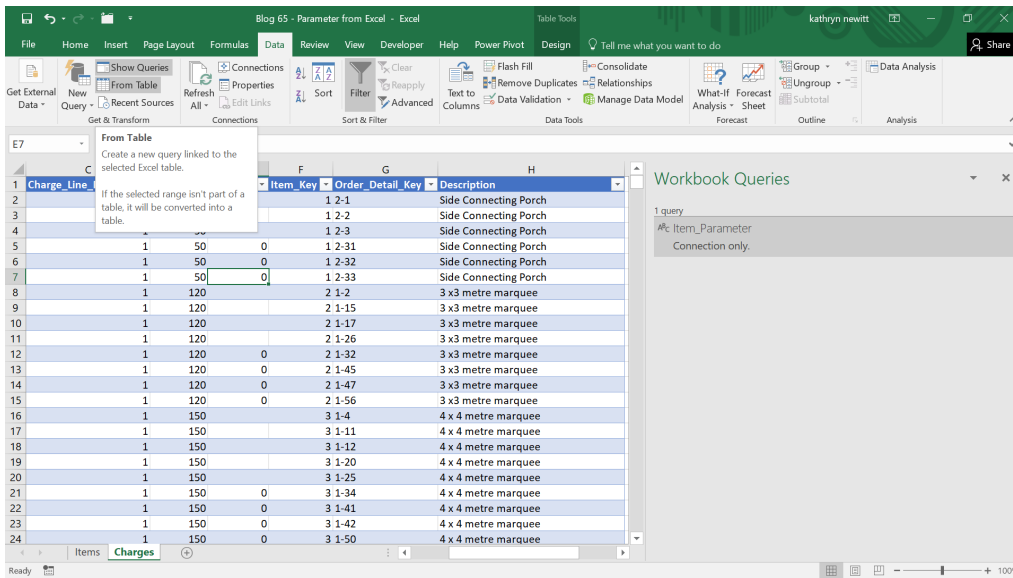
In a new Blank Query, we'll enter the following M function:

```
Excel.CurrentWorkbook() { [ Name="Parameter_Range" ] } [ Content ] { 0 } [ Column1 ]
```

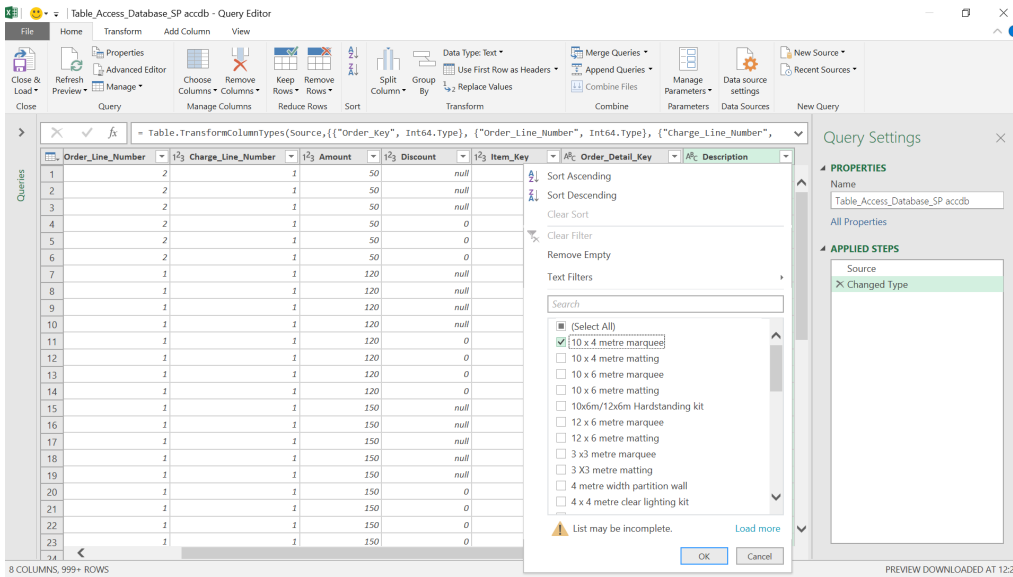
This will extract the value from the cell. We'll rename the new query 'Item_Parameter' and save it as a 'Connection Only' query.



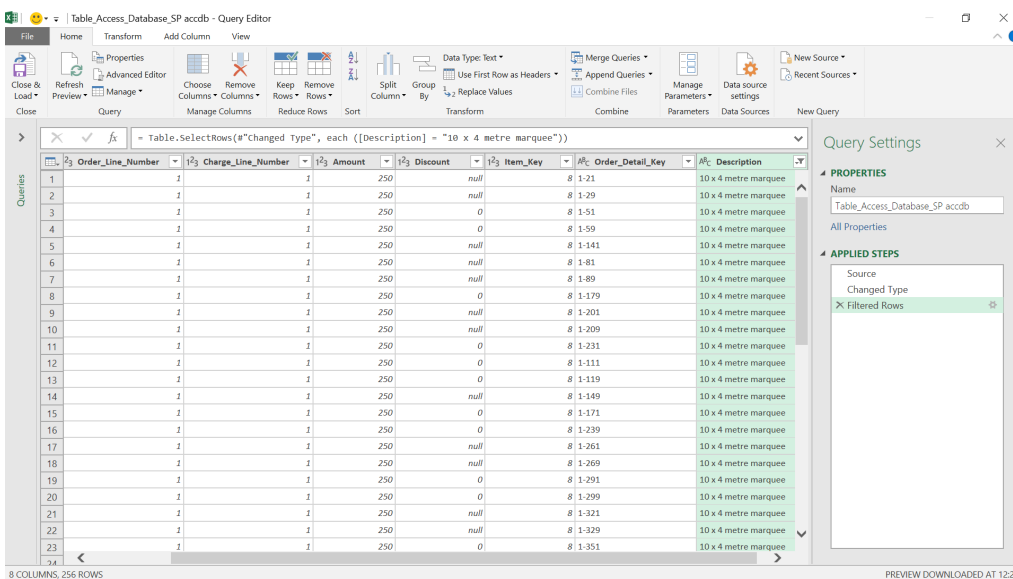
Now we need to create a query on 'Charges', which will select charges that match a particular item. We may do this using the GUI options available. Let's create a new query 'From Table' whilst in the 'Charges' sheet:



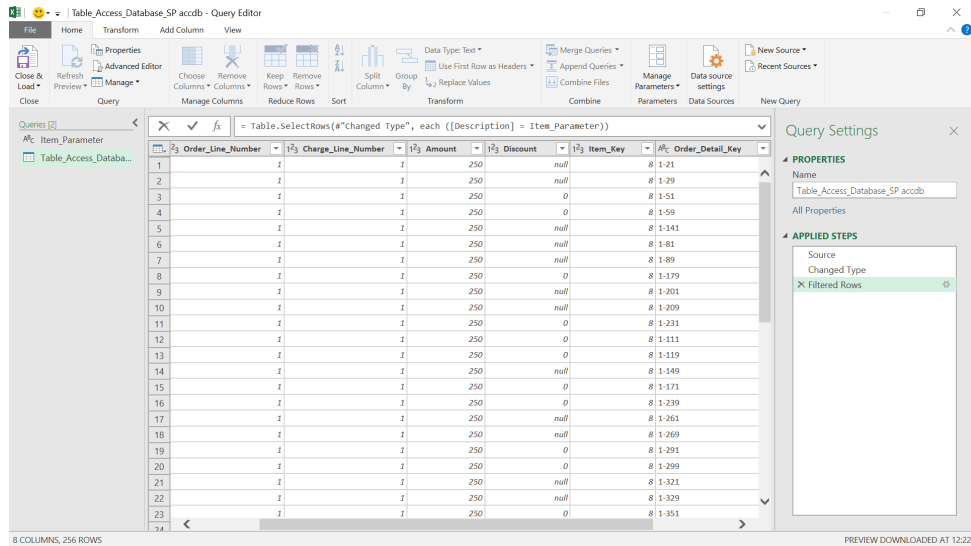
We may then choose to filter on **Description**:



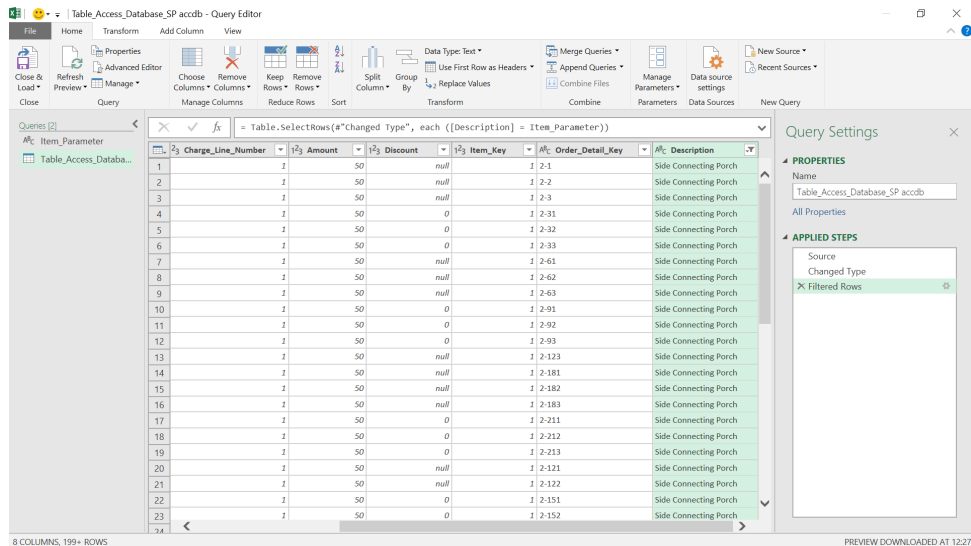
It doesn't matter which description we pick at this stage – we're simply getting Power Query to create the step for us.



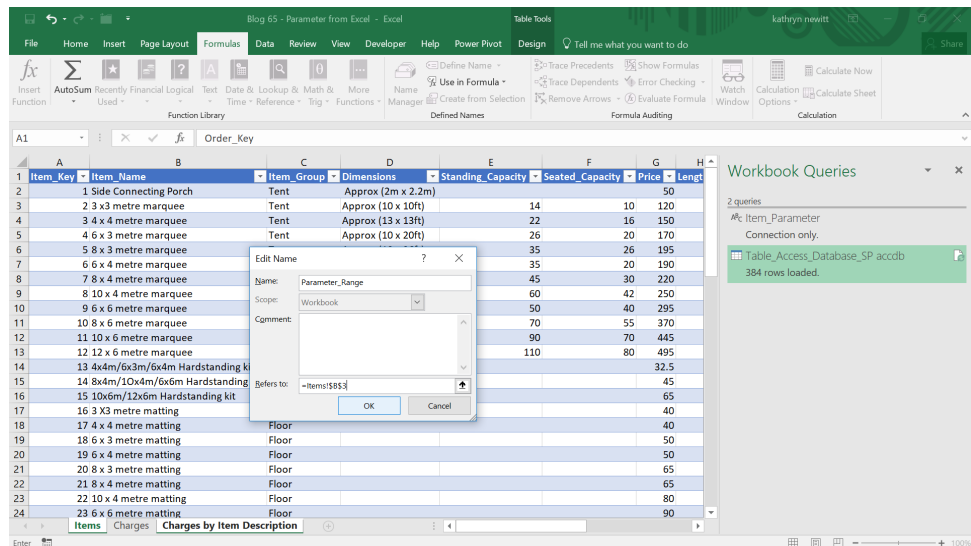
Now we have this step, instead of picking those charges that are pertaining to '10 x 4 metre marquee' we wish to use a parameter instead. If we expand the 'Queries' pane to the left of the screen, we can make sure we have the correct name.



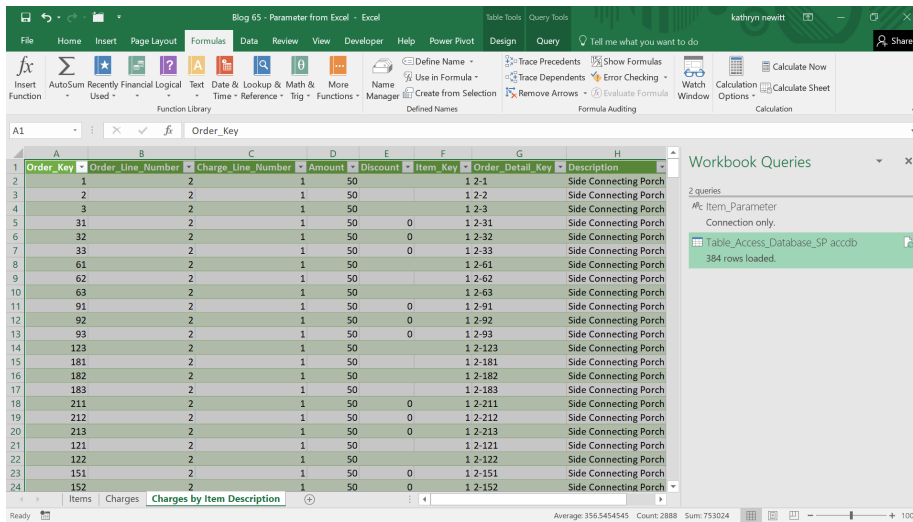
We then select the tick icon (left of the Formula bar) to change the step to use my parameter.



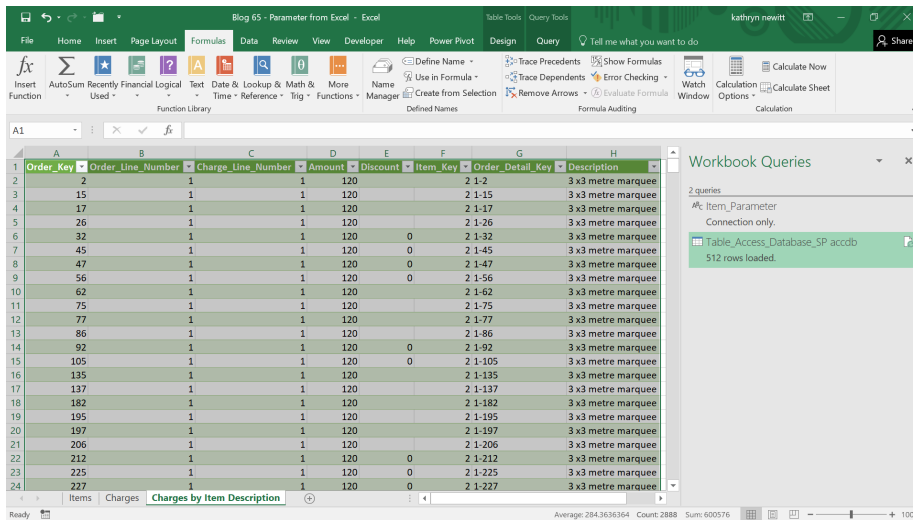
The correct data is shown. We can update the parameter (by changing the cell or editing the named range) in our Excel worksheet and refresh the query to change the data that is selected. Let's try this by changing 'Item_Parameter' to point at the next item description.



Before we refresh our query, we note that the original data appears:



Once we refresh the query, the new parameter value is used.



More next month!

Power BI Updates

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

This month, let's look at a simple scenario where we may use a parameter in Power Query which is populated from a cell in my Excel workbook. Below, I have a spreadsheet with charge data pertaining to items in my fictitious inventory:

Reporting

- New Format pane Preview
- Page and Bookmark navigators:
 - o Page navigator
 - o Bookmark navigator
 - o Formatting options
- Sort legend
- Bold / italic / underline for text formatting
- Text box super / subscript and bulleted lists
- Pie and Donut chart rotation
- Scorecard visual
- Update on tenant-level Bing Maps feature switch

Data Connectivity and Preparation

- Azure Synapse Analytics (new connector) in Preview
- Google Sheets (new connector) in Preview
- Azure Cosmos DB V2 (new connector) in Preview
- Delta Sharing (new connector) in Preview
- Google BigQuery (connector update)
- Cognite Data Fusion (connector update)
- Dremio Cloud (connector update)

Service

- Administration and governance:
 - Dataset hub improvements
 - Azure Lockbox Support for Power BI
- Power BI Goals:
 - Goal level permissions
 - Power Automate Integration
 - Moving goals around the scorecard
 - Showing / hiding and moving columns

Mobile

- Use QR codes with URL parameters (Android and iOS)
- Updated report filters (Android and iOS)

New Format pane Preview

If you have ever spent (considerable) time guessing and testing what various format settings do, you might be pleased to know that this update sees a redesigned Format pane! Apparently, Microsoft has spent the past few months going card by card, property by property to build

Embedded Analytics

- Power BI Embedded Gen2 now Generally Available
- Control embed token expiry
- Support embedding a Power BI report that contains a paginated report visual
- Explore your report in the developer playground

Visualisations

- New visuals in AppSource
- Find report sample files easily in AppSource
- Inforiver standard edition by Lumel
- Drill Down Map PRO by ZoomCharts
- Lipstick Bar Chart by Nova Silva
- Charticulator (version 1.1.2)

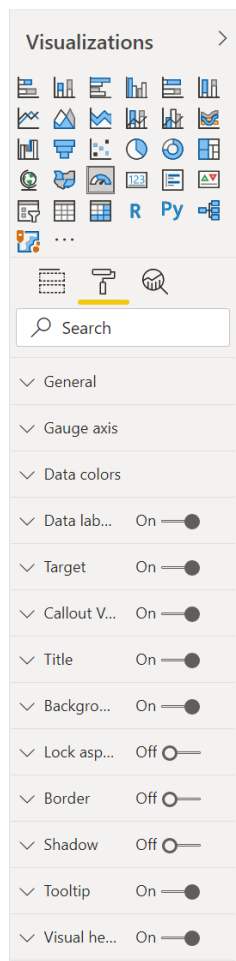
Other

- Automatic Page Refresh now supports DirectQuery for Power BI datasets and Azure Analysis services.

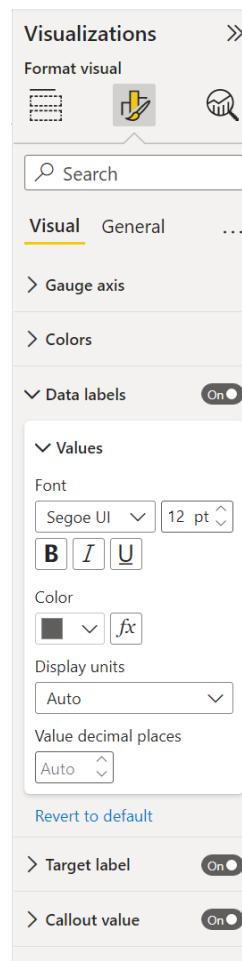
Let's go through each in turn.

a more consistent and easier to use formatting experience. You can see for yourself how it's worked out by turning on the preview switch: **File -> Options and settings -> Options -> Preview Features -> New Format pane.**

Old Format pane



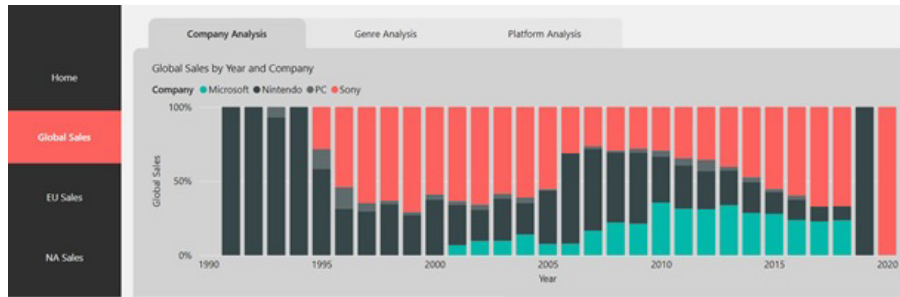
New Format pane



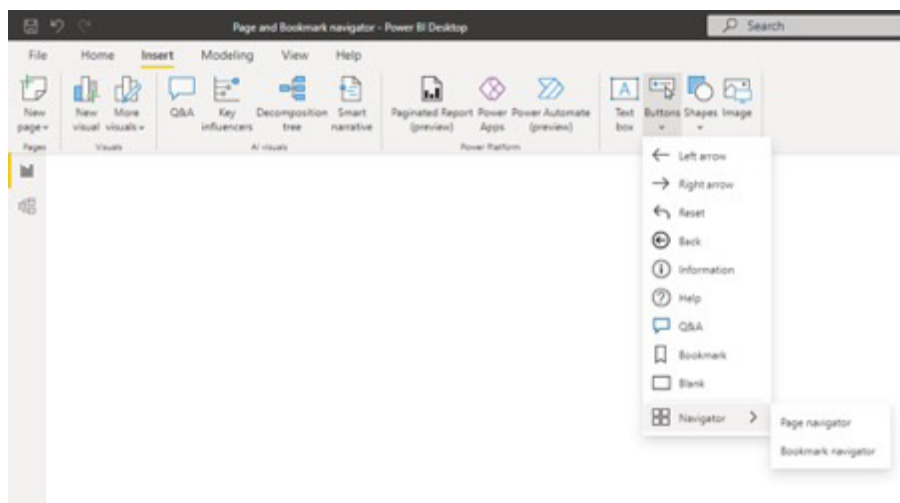
Page and Bookmark navigators

Previously, setting up a custom Page or Bookmark navigation experience in your report could be pretty time consuming because it required creators to setup each individual button for every Page or Bookmark and duplicate each button if they wanted to show a selected state. In this month's

release, two new built-in navigators have been added to allow creators to quickly build Page and Bookmark navigation experiences with just a few clicks.

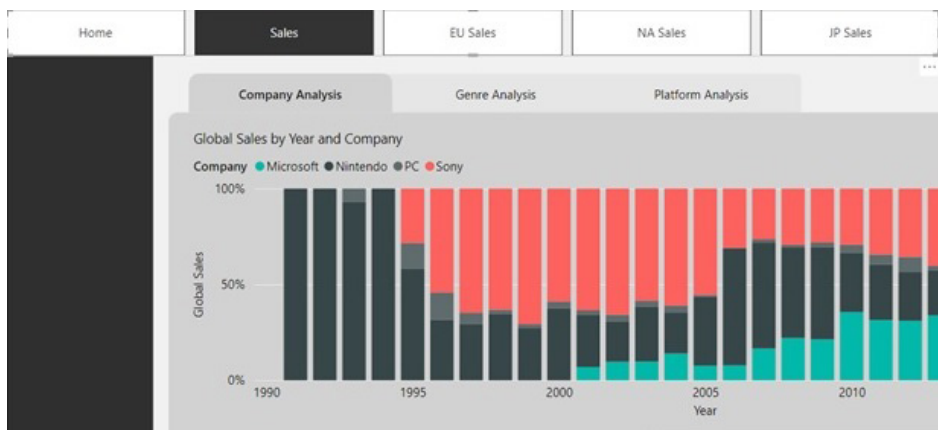


These new navigators should save creators hours of effort building and managing their page or bookmark navigation experiences. You can find this new capability in Power BI Desktop within the **Insert tab -> Buttons -> Navigator**:



PAGE NAVIGATOR

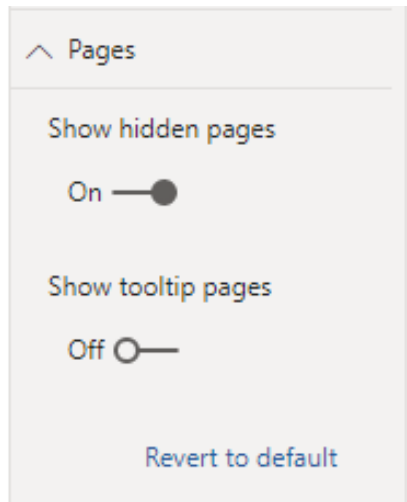
When you select the Page navigator option, a Page navigator will be created for you automatically:



The Page navigator is automatically synchronised with your report Pages, meaning:

- titles of the buttons match the Page display names
- ordering of the buttons match the order of your report Pages
- the selected button is the current Page
- as you add or remove Pages in your report, the navigator will automatically update
- as you rename Pages, the titles of the buttons will automatically update.

If you want further customisation of which Pages to show or hide in the Page navigator, you can go to the **Format navigator pane -> Pages** tab:

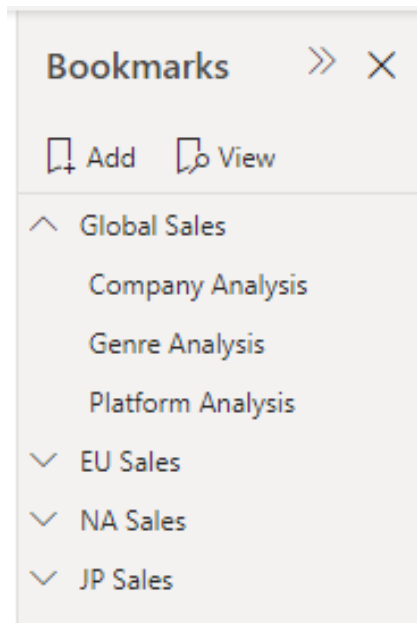


You have the option to 'Show / hide hidden pages' or 'Show / hide tooltip pages'.

You should note that if you're testing out the Page navigator in Desktop or Edit mode of the Service, you'll need to **CTRL + Click** to navigate to the desired page.

BOOKMARK NAVIGATOR

Before you create the Bookmark navigator, you need to make sure you already have Bookmarks created. Additionally, you should create separate Bookmark groups if you plan on creating different Bookmark navigators within the same report.



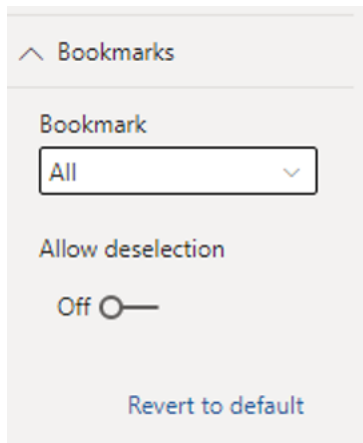
Once you've created your Bookmarks and select the Bookmark navigator option, a Bookmark navigator will be created automatically for you:



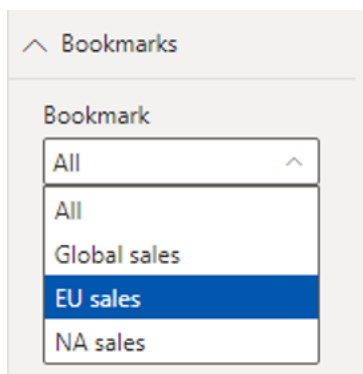
Similar to the Page navigator the Bookmark navigator will synchronise automatically with your report Bookmarks, meaning:

- titles of the buttons match the Bookmark display names
- ordering of the buttons match the order of your report Bookmarks
- the selected button is the last selected Bookmark
- as you add or remove Bookmarks in your report, the navigator will automatically update
- as you rename Bookmarks, the titles of the buttons will automatically update.

If you require further customisation of which Bookmarks to show or hide in the Bookmark navigator, you can go to the **Format navigator pane -> Bookmarks** tab:

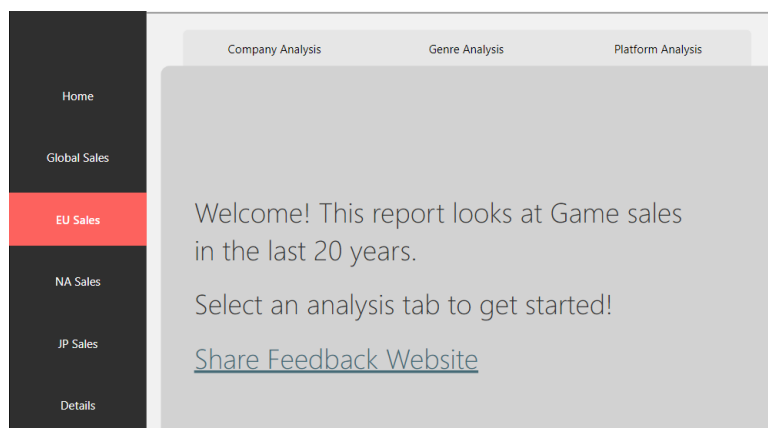


By default, **All** Bookmarks will be shown in the Bookmark navigator. However, you can create and select a specific Bookmark group to only show the Bookmarks within that group.

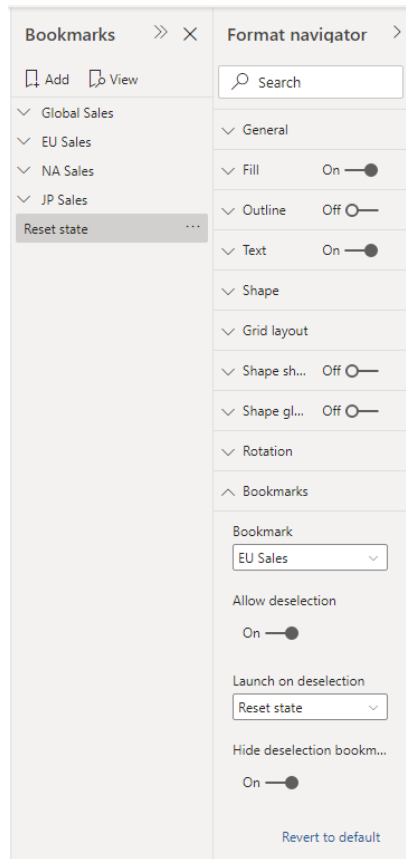


You also have the option to **Allow deselection**, meaning users may unselect the selected button in the Bookmark navigator. This option is useful for building a toggle-like experience or allowing for an unselected

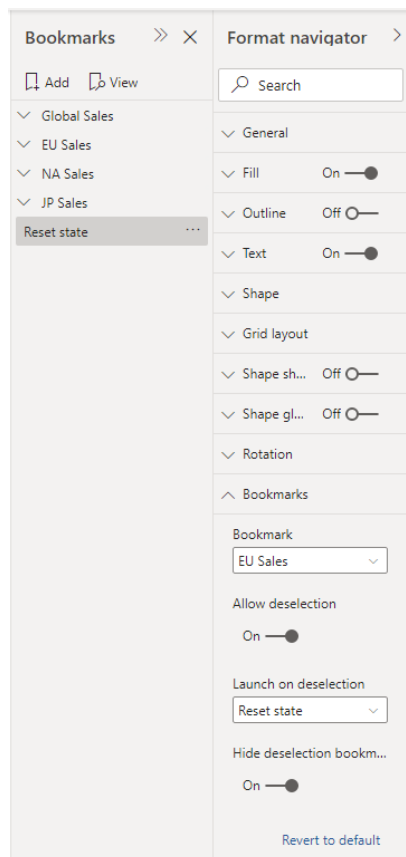
default state. To set up either of these types of experiences, you'll need to first Bookmark the desired unselected state. Here's an example of an unselected state:



Once you have the unselected state Bookmarked, you'll need to turn on **Allow deselection** and select the Bookmark that you want to **Launch on deselection**:



If the Bookmark that you're using for deselection is within the Bookmark navigator already, you can choose to **Hide the deselection Bookmark** within the navigator if you don't want to show it:



FORMATTING OPTIONS

Just like other buttons in Power BI, there are numerous formatting options for the navigators including:

- Fill
- Text
- Outline
- Shape
- Shape shadow
- Shape glow
- Rotation.

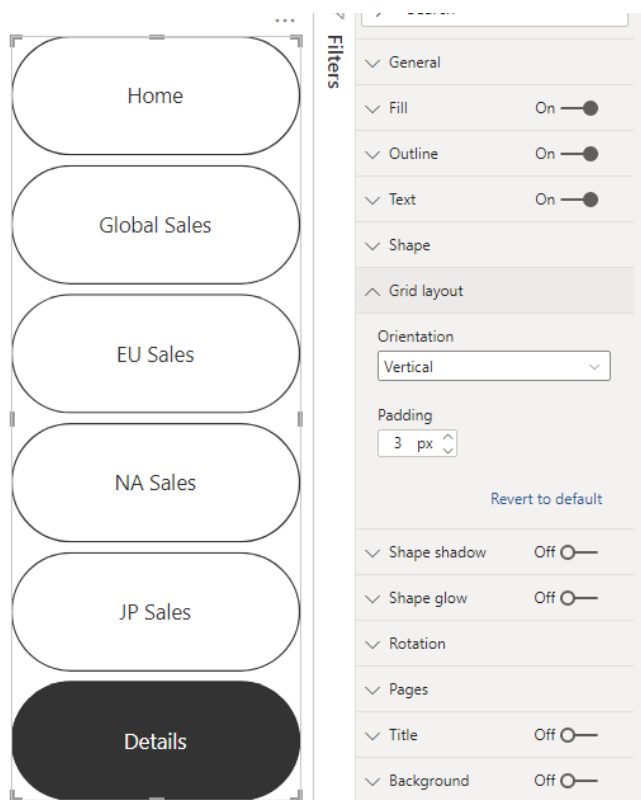
The navigators also include two additional formatting options:

- Grid layout
- Selected state.

Grid layout

Grid layout tab includes options to change the **Orientation** of the navigator:

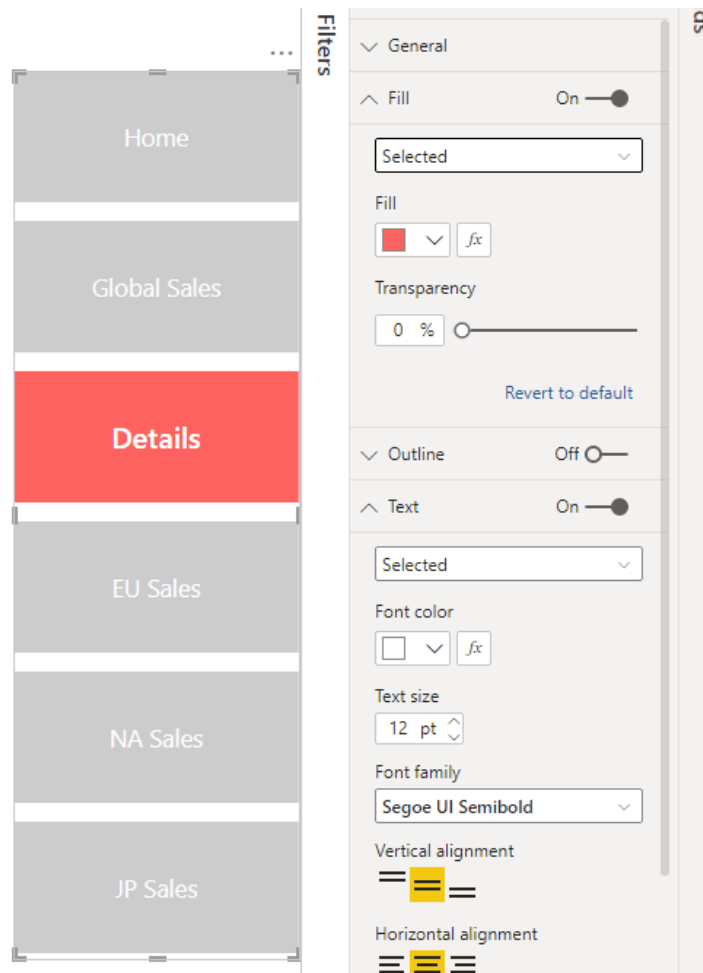
- Horizontal
- Vertical
- Grid



It also includes the option to change the **Padding** between buttons in the navigator.

Selected state

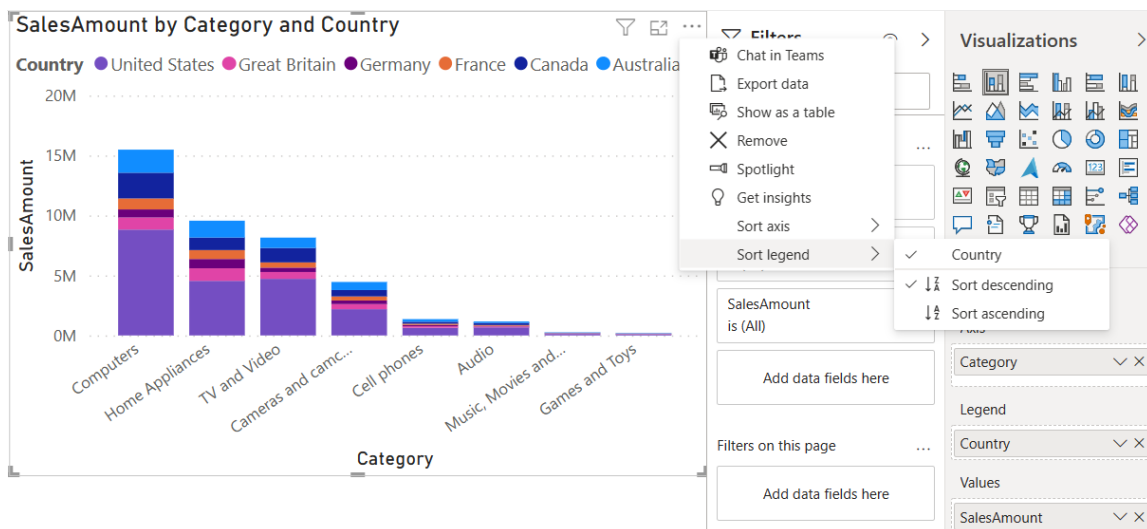
The navigators have the option to customise the 'Selected state' of the button. You can use this option to help the Selected state of the button stand out from the default state. In this example, the Fill and Text formatting have both been customised for the Selected state:



Sort legend

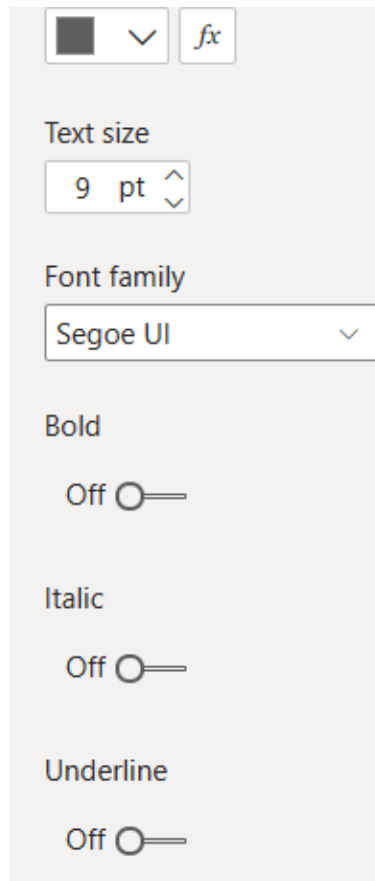
It's now possible to adjust the sort direction for items in legends. To control these, click on the context menu of the visual header, and select the field and sort direction in the **Sort legend** flyout. This will also help to reorder some of the data in visuals like stacked column charts, as well

as which lines or areas are rendered on top of others in other Cartesian charts. It should be noted that currently this feature only allows you to sort by the field in your **Legend** field well.

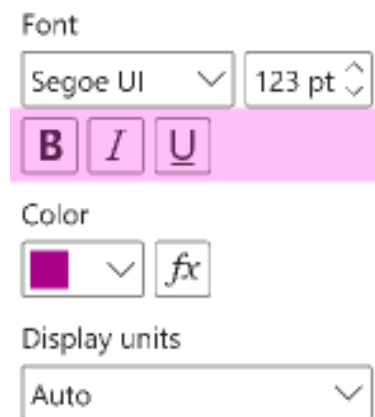


Bold / italic / underline for text formatting

Wherever you see text formatting options like font family, font size and colour, you will now find new options to allow you to bold, italicise and underline that text as well. In the old Formatting pane, this option would have three toggle switches:



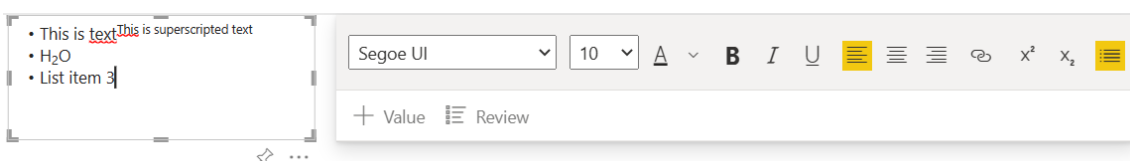
In the new Formatting pane, you'll see them as a familiar set of icons, all on one line:



These new options should show up in the Formatting pane for text wherever you are in your Power BI report.

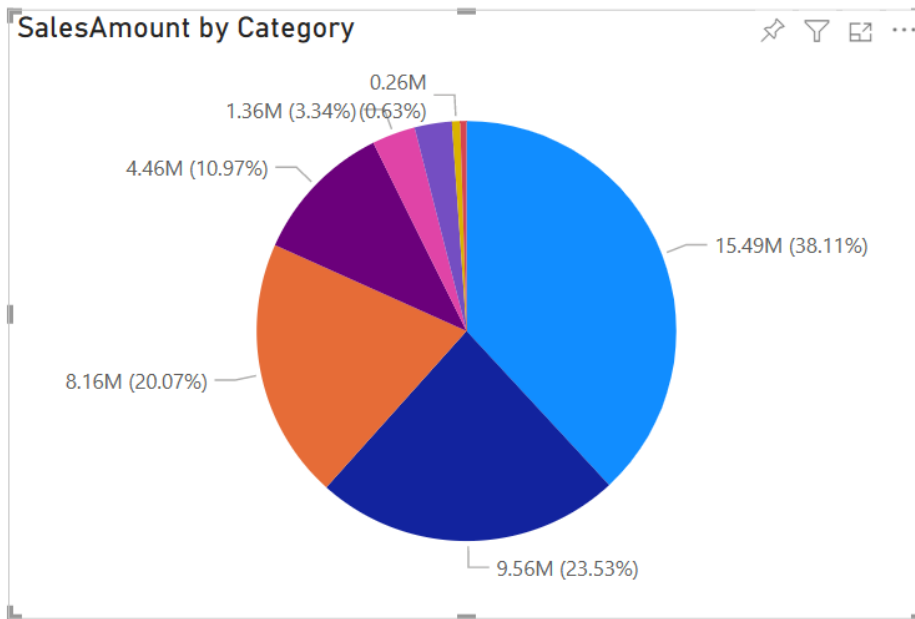
Text box super / subscript and bulleted lists

The text box has now been updated with three new formatting options: superscript, subscript and bulleted lists. You can access them through the floating Formatting pane.



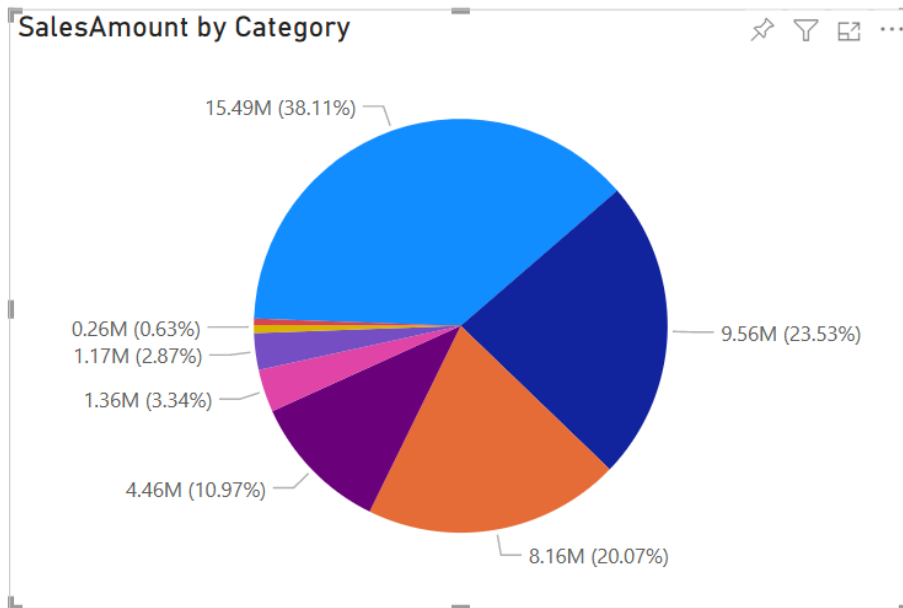
Pie and Donut chart rotation

Currently, Pie and Donut chart slices are drawn in clockwise order, beginning at the 12 o'clock position, which often results in the final slices being drawn near that first large value, approaching it clockwise from the opposite side. Those final slices are often smaller values, and thus smaller slices.



While this creates an intuitive Pie chart, it leaves less room for data labels describing the smaller slices, as text runs horizontally but the slices are vertical. To assist, Power BI has introduced a new **rotation** option in the

Formatting pane to provide greater control over where you start drawing the Pie. You may find that in the Shapes card in the old Formatting pane as 'start angle' and in its own Rotation card in the new Formatting pane.

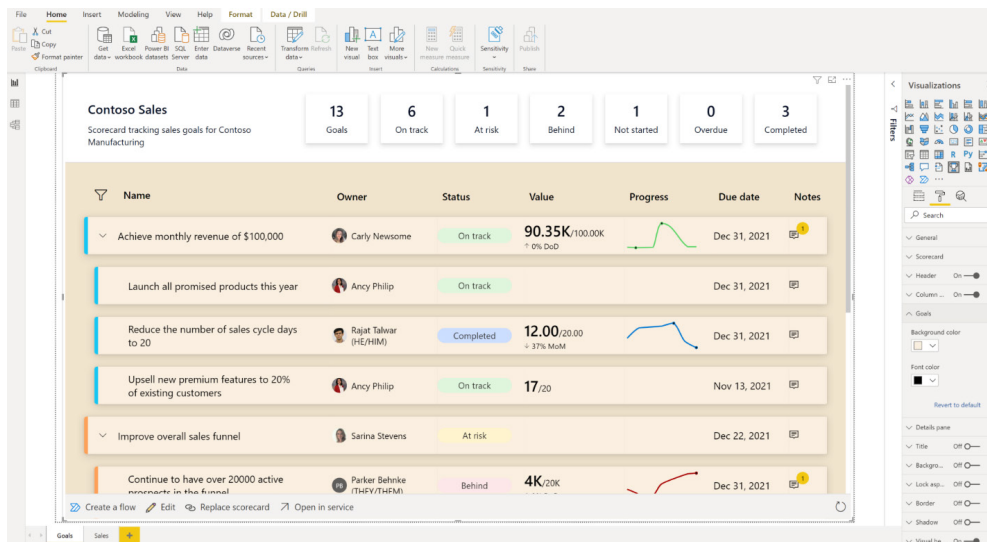


Scorecard visual

Integrating Goals and Scorecards is a key aspect of Power BI reporting. A new Scorecard visual has now been added which can be attached to Power BI reports. When included in a report, these visuals let you see the entire Scorecard and even make updates to your Goals.

Goals may also be created from within Power BI Desktop now, streamlining

how they are created and managed. The underlying Scorecard added to a report must be added to the Power BI Service, but it's straightforward to do this from Desktop. The visual supports various formatting options for Scorecards, ranging from font style and colours to backgrounds, so users may modify existing Scorecards.



Update on tenant-level Bing Maps feature switch

Last month, Power BI introduced a tenant-level feature switch to enable or disable the Bing Maps visual. Along with this change, Microsoft indicated that Bing Maps moving forward will require explicit opt-in for new tenants and will be disabled by default. The Desktop constraint for Bing Maps

is forthcoming but not yet implemented, and as a result, there may be situations where report viewers in the Power BI Service cannot see a maps visual that its author was able to see when creating the report in Power BI Desktop. Watch this space.

Azure Synapse Analytics (new connector) in Preview

This update provides the launch of the Public Preview of the Azure Synapse Analytics Workspace (Beta) connector for Power BI Desktop that enables users to work with all Synapse Databases. The connector automatically discovers Synapse Workspaces that users have access to (via ARM). With the support for related tables and auto import of relationships from Synapse to Power BI Desktop, users may now use their Synapse data to improve update times.

For scenarios which need leveraging features such as DirectQuery and native query on the SQL DBs, you should continue to use the existing Synapse Connector (Azure Synapse Analytics (SQL DW)), as these functionalities aren't supported in this Public Preview version of Azure Synapse Analytics workspace (Beta) connector presently.

Google Sheets (new connector) in Preview

Connecting to Google Sheets is the most frequently requested connector-related idea. Microsoft has heard you. The Google Sheets connector is built on top of the Google Sheets API and enables out-of-box connectivity to Google Sheets. After selecting 'Google Sheets' in the Power BI 'Get Data' menu, you will be prompted to enter a Google Sheet URL:



You simply copy and paste in a Google Sheet URL that you have access to into the input and select OK. Authentication for the connector is performed through standard Google OAuth. When you select 'Sign in', an external browser window will open, and you will need to log in to your Google Account and give permission for Power BI Desktop to access your Google Sheets. Once authenticated, you will be shown the Navigator

experience showing all the individual Sheets within the specified Google Sheet.

You may select the Sheet(s) to import to the Power Query Editor and continue to transform your data there. The Google Sheets connector supports access to Shared Drives and will require you to authenticate to every Google Sheet you connect to.

Azure Cosmos DB V2 (new connector) in Preview

The Azure Cosmos DB V2 connector was supposed to have been released in this update, but there appears to be confusion whether this is indeed the case. On its Power BI website, Microsoft has recommended “... continuing to use the Azure Cosmos DB V1 connector”.

If this is the case, to query data stored in the transactional store of Azure Cosmos DB containers, users should utilise the native Power BI connector referred to as Azure Cosmos DB V1 connector) that is Generally Available,

but this only supports Import mode. The upcoming new native Power BI connector (known as Azure Cosmos DB V2 connector) will support querying the transactional store in both DirectQuery and Import modes. In addition to supporting DirectQuery mode, the V2 connector will include performance optimisations related to query pushdown and data serialisation too.

Delta Sharing (new connector) in Preview

Delta Sharing is an open protocol that enables secure exchange of datasets across products and platforms by leveraging proven and scalable technologies such as REST and cloud storage.

The Power BI Delta Sharing connector is an implementation of a Delta Sharing client, which can connect to any Delta Sharing server so users can discover, analyse and visualise datasets shared with them via the Delta Sharing protocol.

To Power BI users, the connector is a route for gaining visual insights on live datasets shared with them through Delta Sharing and leveraging the rich and familiar functionality of Power BI with Delta Shares. To data providers, the connector opens up their data products (as distributed via Delta Sharing) for secure business intelligence to any Power BI user.

Google BigQuery (connector update)

This release includes performance improvements for the Google BigQuery connector. The number of metadata calls in the connector required to load the Navigator experience has been reduced, so you should see improved loading times.

Cognite Data Fusion (connector update)

This update has fixed multiple issues related to custom queries and the ‘Organizational Account’ (sic) login. Further, Microsoft has also remedied the issue with aggregating time-series when using tags containing special characters.

Dremio Cloud (connector update)

There is now additional support to connect to regions in Dremio Cloud. Region is supported as a drop-down box and is mandatory. The Project identity is now changed to optional. All other features remain the same.

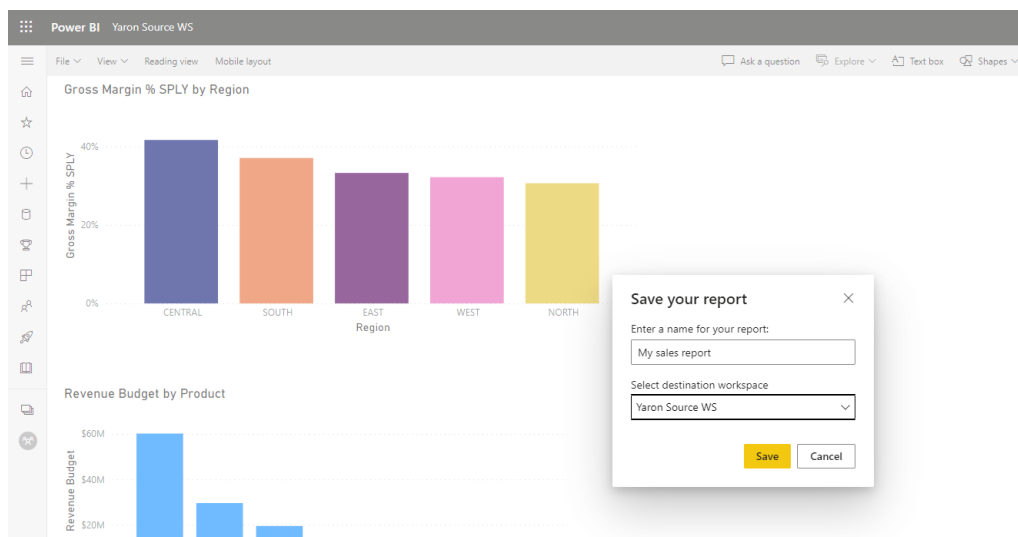
Administration and governance

DATASET HUB IMPROVEMENTS

Select a Destination Workspace When You Create a New Report Online

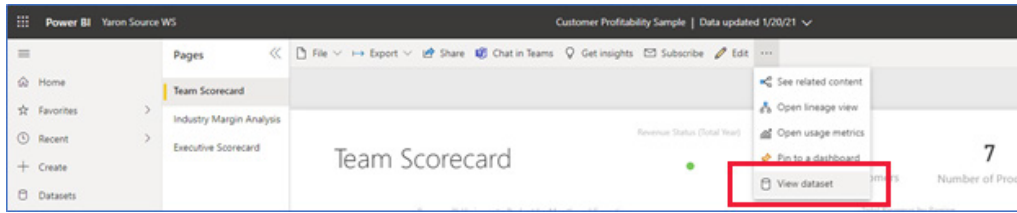
In the datasets hub, you may perform a variety actions on your datasets, one of which is to create a new report. With this release, after you’ve created your report visuals, once you hit the save button you may now choose which workspace to save the report to. This is an improvement

for the functionality of shared datasets, as it provides you with the ability to save reports created online in a workspace other than the workspace where the dataset resides.



Quick Access to the Dataset Page from Report View

Power BI has now added quick access from the Report view to the underlying dataset.



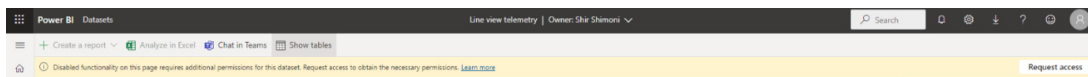
The dataset page can be useful for multiple users and scenarios. For example, if you are the dataset owner, you may:

- quickly check the dataset's refresh status and trigger a manual refresh
- quickly check lineage to troubleshoot data update failures
- view impact analysis and notify contacts about any problem(s) with the data
- quickly see usage metrics (unique viewers and number of views) for the report you opened the dataset page from, as well as for other reports that are built on top of this dataset.

If you are not the dataset owner, you can still benefit from:

- checking the dataset's refresh status
- viewing other reports that use the same dataset
- being able to create a new report on top of the dataset.

If you don't have Build permission on the dataset yet, you will see a 'Request access' button you can use to request it. Once you've obtained Build permission on the dataset, you can start creating your new reports on top of it.



AZURE LOCKBOX SUPPORT FOR POWER BI

Most operations, support and troubleshooting performed by Microsoft personnel do not require access to customer data. In those rare circumstances where such access is required, Customer Lockbox for Microsoft Azure, now supported in Power BI, provides an interface for customers to review and approve or reject customer data access requests. It is used in cases where a Microsoft engineer needs to access

customer data (for example, they need to remote into a server and take a memory dump of a Power BI dataset in import mode). This provides you with the assurance that any access by the live site or engineering teams to your content as part of a standard operating procedure will go through your administrator's approval procedures.

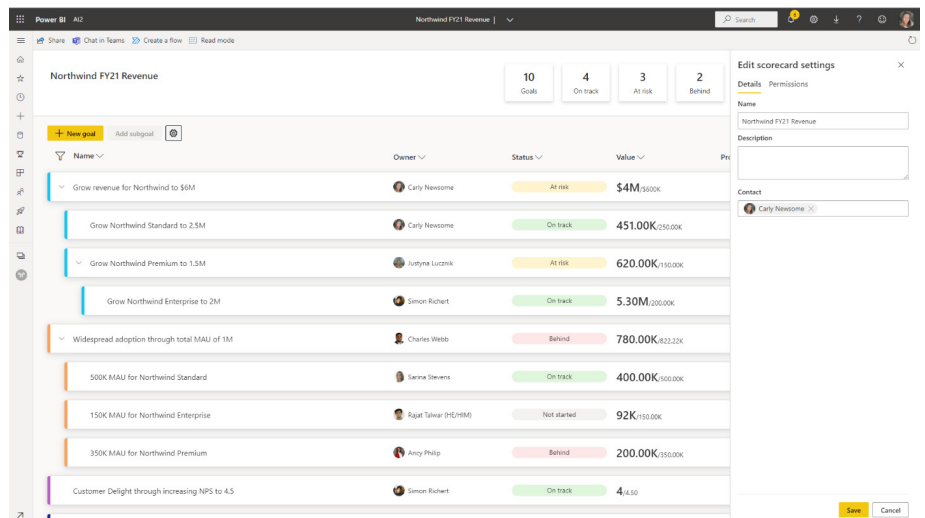
Power BI Goals

GOAL LEVEL PERMISSIONS

In many organisations, people in different roles should have access to view and edit specific Goals. For example, it might be only managers should see Goals related to human resources and finances, whereas all employees may view Goals related to operations. In many cases, specific individuals or groups should be the only ones who can edit or update Goals. Using Goal level permissions, these scenarios are easily accomplished.

Goal level permissions let you set specific view and update permissions at the Goal level. Here's how to get started setting up goal level permissions:

- ensure you're in Edit mode for the Scorecard
- open the Settings pane
- open the new Permissions tab.



On the new Permissions tab, you can create roles with different permission and assign specific user groups to those roles. For example, to give a view only experience to the leadership team, simply create a view only role, then assign the security group for my leadership team to that role. Similarly, you may give view and update permissions to individual sales representatives to ensure that they can always keep their Goals up to date.

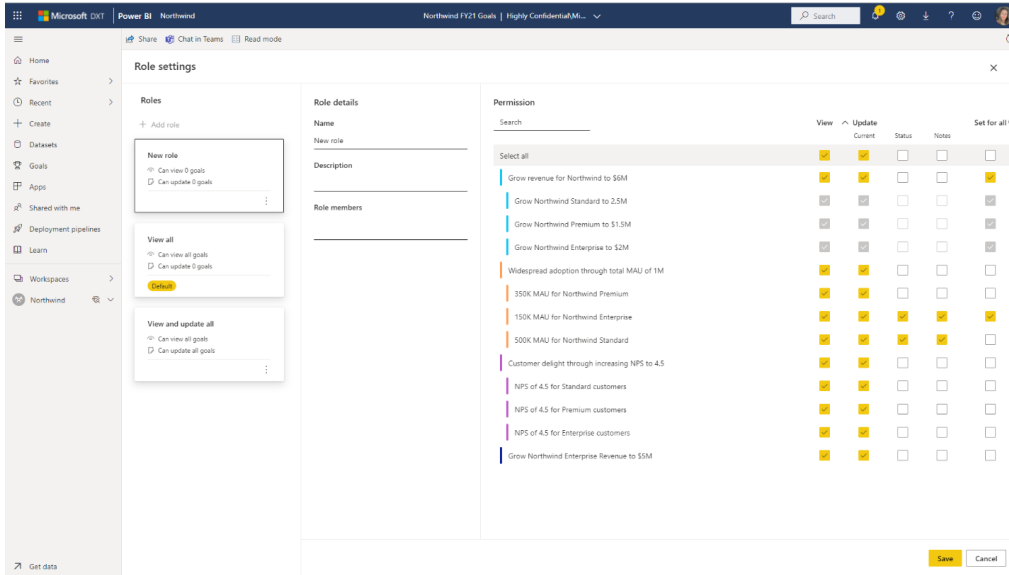
It should be noted that there are a few different types of permissions:

- **View permissions:** let users view specified goals in a Scorecard
- **Update permissions:** let users update specific aspects of a Goal.

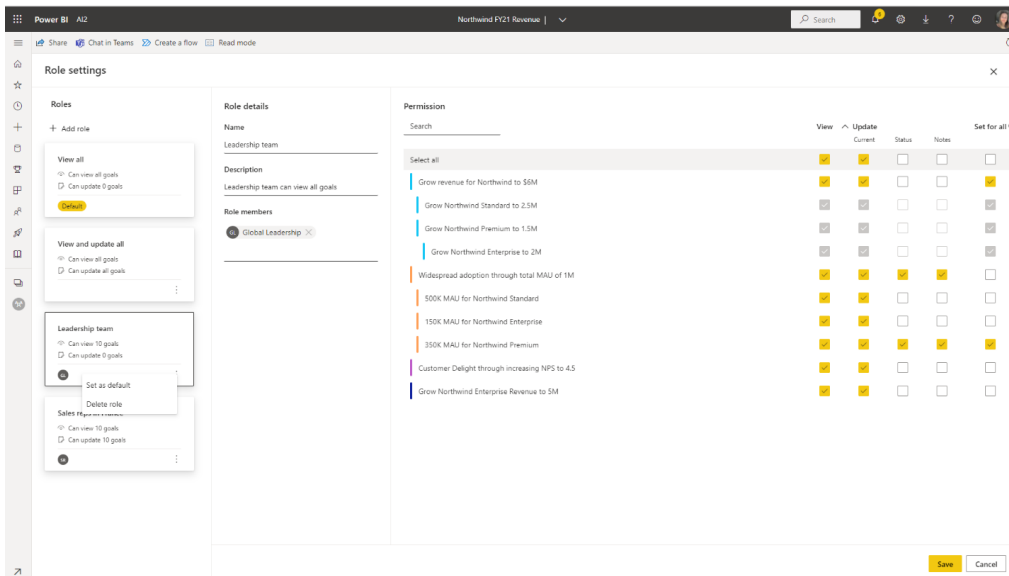
There are a few options available under update permissions, and you can select any combination of the following:

- o **Note:** users May add notes in a check-in
- o **Status:** users may update status in a check-in
- o **Current Value:** users may update the current value in a check-in.

Additionally, selecting the ‘Set for all’ option, turns on inheritance for all existing and future subgoals under that level. This means future new goals will adopt the permission set you selected.



Another aspect of Goal level permissions is enabling a “default” permission model that’s applied to anyone accessing the Scorecard. You may create a role with any combination of permissions you choose, and make that the default permissions, ensuring that any time anyone lands on the Scorecard, they’re seeing exactly what the Scorecard author selects.



Goal level permissions also apply to the underlying scorecard dataset generated. For example, if you were to give the sales team view access to five Goals on the Scorecard, those are the five goals they will see in the underlying dataset as well. The support for automatic roles based on Goal level permissions (full Row Level Security support) is coming, but this be contained in this Goal level permissions feature.

By enabling Goal level permissions, you ensure that your Scorecard users are accessing only what they should be able to see and updating only what they should be able to update.

POWER AUTOMATE INTEGRATION

Many organisations want to use Scorecards as part of processes that help them achieve results faster. However, manually monitoring Scorecards may be resource intensive and error prone.

This is where the new Power Automate integration for Power BI Goals comes in. This new capability helps you automate business processes

when important changes happen within your Scorecard. It helps you and your colleagues to quickly respond to changing conditions by keeping everyone up to date and taking automated actions to improve outcomes. This new capability is very easy to use because you can launch Power Automate directly from your Scorecard and immediately construct your automated flow.

To do this, Power BI has enabled a set of triggers and actions within Power Automate:

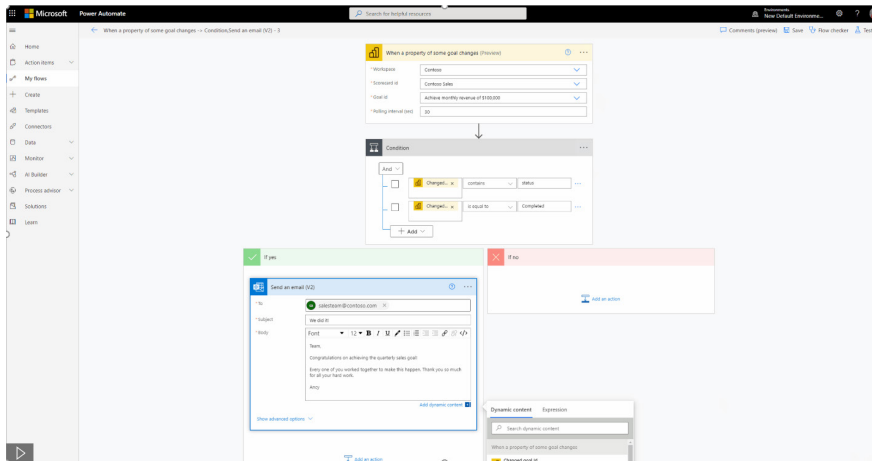
- Triggers:
 - o when a Goal changes (e.g. status, owner)
 - o when someone adds or edits a check-in
 - o when an owner is assigned to a Goal
 - o when a data refresh for a Goal fails
- Actions:
 - o create a Goal
 - o create a check-in
 - o add a note to a check in
 - o create a Scorecard
 - o update a check-in
 - o update a Goal
 - o get Goal(s)
 - o get Goal check-in(s).

In addition to these actions and triggers, there are also some new templates that will be rolling out within the next few weeks. These templates will allow you to choose a flow that hopefully will align with your more complex business scenarios and ensure that you have the building blocks you need to automate your process. Such templates will include:

- triggering a teams notification when a status changes to “behind” or “at risk”
- sending reminders to team members at a specified interval with a link to a Scorecard or specific Goals to review
- Notifying a specific team member when they are assigned to a new Goal and should perform a check-in
- Sending a forms survey that gets added as a check-in note on a Goal at a specified interval
- Sending a congratulations email when a team completes a Goal.

Using Power Automate with your Power BI Goals should help you respond more quickly to changing conditions and to use data to take better actions.

Name	Owner	Status	Value	Progress	Due date	Notes
Grow revenue for Northwind to \$6M	Carly Newsome	At risk	\$4M _{150.00K}		Nov 21, 2021	
Grow Northwind Standard to 2.5M	Carly Newsome	On track	451.00K _{150.00K}		Nov 21, 2021	
Grow Northwind Premium to 1.5M	Katyna Luczek	At risk	620.00K _{150.00K}		Nov 21, 2021	
Grow Northwind Enterprise to 2M	Simon Richert	On track	5.30M _{200.00K}		Nov 21, 2021	
Widespread adoption through total MAU of 1M	Charles Webb	Behind	780.00K _{202.22K}		Nov 21, 2021	
500K MAU for Northwind Standard	Sanna Stevens	On track	400.00K _{100.00K}		Nov 21, 2021	
150K MAU for Northwind Enterprise	Rajat Tavar (H/HRM)	Not started	92K _{150.00K}		Nov 21, 2021	
350K MAU for Northwind Premium	King Philip	Behind	200.00K _{150.00K}		Nov 21, 2021	
Customer Delight through increasing NPS to 4.5	Simon Richert	On track	4.450		Nov 21, 2021	

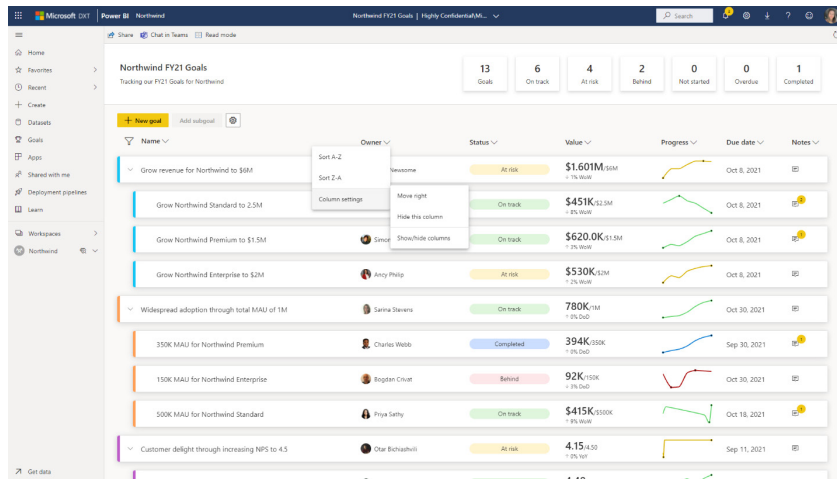
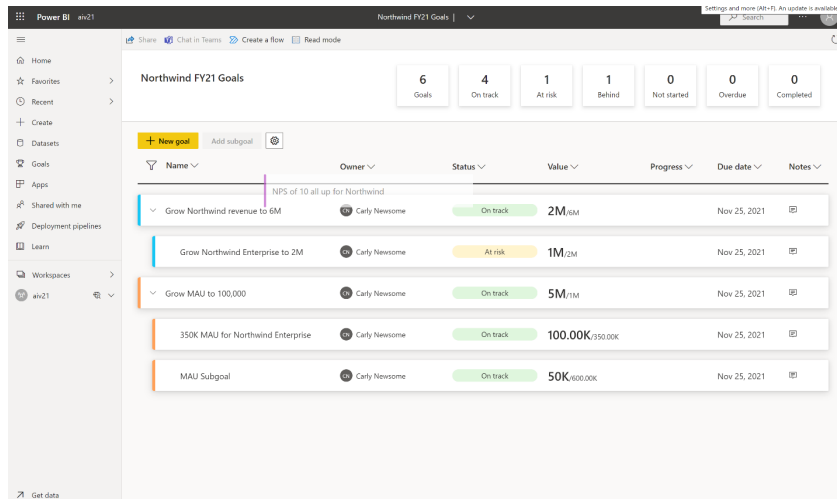


Coming very soon!

MOVING GOALS AROUND THE SCORECARD

As things change within your business, so you may require the ability to drag and drop Goals around the Scorecard, making it easier to maintain an accurate view of your company's priorities and Goal groupings.

To do this, simply click the Goal you want to move and drag it to the specified location. You may reorder Goals within a given family, move Goals entirely to another family, or make any Goal a standalone Goal.



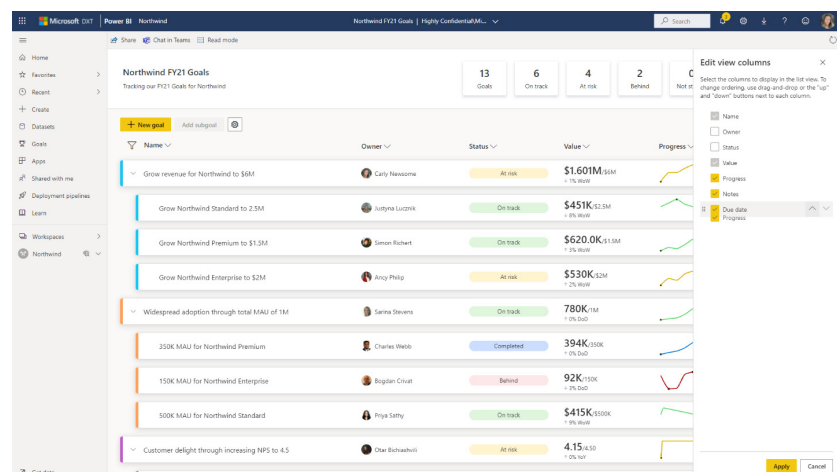
Reordering Goals makes it easier to ensure the right priorities are reflected and your Scorecard order is appropriately modified.

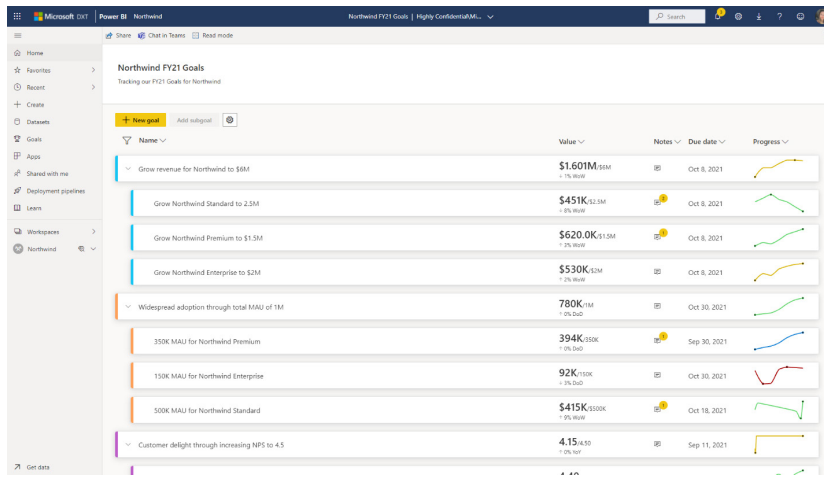
SHOWING / HIDING AND MOVING COLUMNS

Some organisations may use all of the fields Power BI supplies in a Goal, but there are scenarios where not all of the fields provided are applicable. With this release, you have the ability to hide / show columns in the Scorecard. This will allow Scorecard authors to dictate the columns that consumers see, making sure each Scorecard shows only columns that are appropriate to their scenario. Scorecard authors will also be able re-

order the columns so that the Scorecard is best suited for their audience.

In Edit mode, simply click the arrow icon next to a column and open column settings. Here, you'll be able to specify which columns you want to show or hide. You can also drag the column names up or down to reorder them on the Scorecard.



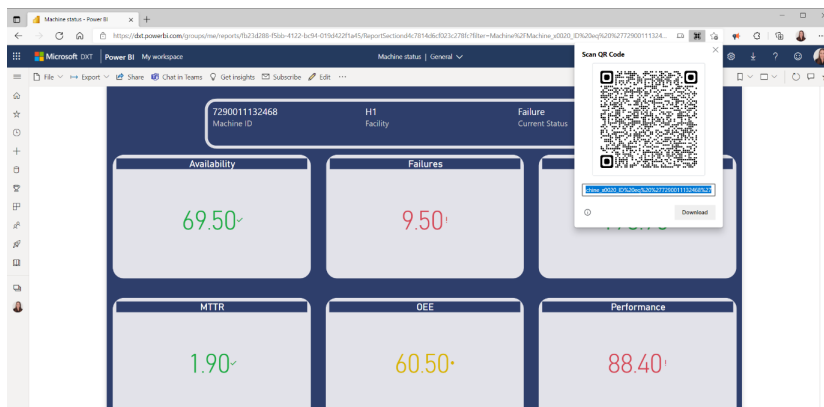


This makes it easier to tailor the Scorecards to your precise needs.

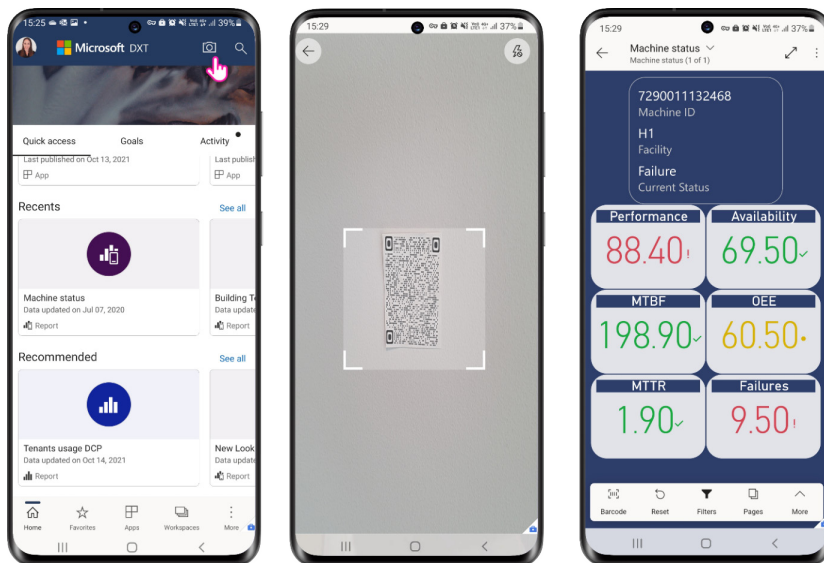
Use QR codes with URL parameters (Android and iOS)

Not everyone appreciates that you may use the Power BI Mobile app to scan a QR code to quickly open a Power BI report. It goes even further now: as of this latest Mobile apps update, you may also add query parameters to the report URL, so that when a QR code with the URL is scanned, those query parameters become the report's filter.

To use this new capability, use a browser based QR generator to generate and save a QR code for the filtered URL (please note that the Power BI service's QR generator doesn't support query parameters).



Open the Power BI Mobile app and tap the camera to scan the QR code. The report will open, filtered according to the URL parameters, viz.

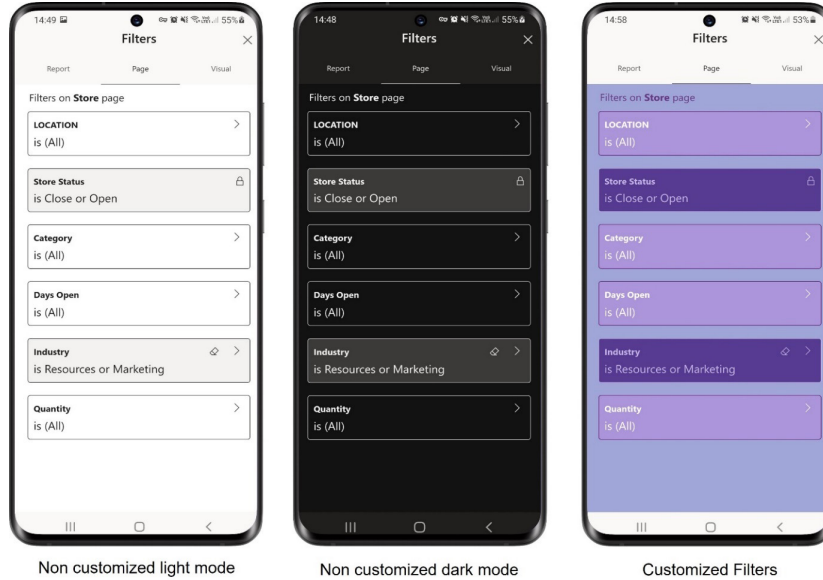


Updated report filters (Android and iOS)

The Report Filter pane has now been redesigned to make it easier to understand the filters applied to Reports and their visuals. The Filter pane revamp includes:

- updated Filter tabs that make it clear what you're filtering, *i.e.* Report, Page or Visual
- highlighting, to indicate which filters are currently being applied
- a cleaner style for the Filter categories and greater ease-of-use for touch gestures.

The new Filter pane also supports Filter colour customisations made by the report creator, so now the Filter pane can have the same look and feel as you've designed, as illustrated below:



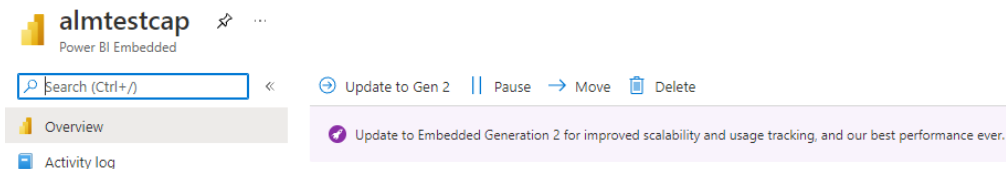
Power BI Embedded Gen2 now Generally Available

Embedded Gen2 has now become Generally Available. Just like Premium Gen2, it gives a performance boost, as cluster nodes are sized like A6 (or P3), and in greater scale, as there are fewer memory restrictions and there's a complete separation between report interaction and scheduled refreshes.

Embedded Gen2 capacities have added values in addition to the benefits

of Premium Gen2: they provide lower entry level for embedded paginated reports and AI workloads, and you can now scale Azure capacities up and down in seconds instead of minutes, without down-time.

API and UI are available to transition Gen1 to Gen2. You can see part of the Azure Resource UI in the screenshot below:



It's important to mention that Gen1 capacities can still be used until auto-migration on Saturday 15 January 2022, but cannot be created using the UI anymore.

It should also be noted that regarding Autoscale, Premium Autoscale is planned for Embedded capacities early in 2022, though customers may still configure vertical Autoscale in Azure, using ARM APIs, Azure Metrics and Alert Rules (whatever all that means!).

Control embed token expiry

This update has added the option to shorten the maximal lifetime of an embed token generated by the GenerateTokenRequest API call by adding the lifeTimeInMinutes parameter to it.

This is useful mainly for security purposes in case your solution requires

you to generate a token that is shorter than the default one, determined by the AAD token expiry time. Keep in mind that this can't be used to extend the default lifetime of the embed token, only to shorten it. This option is also available in the .NET SDK starting from version 3.29.0.

Support embedding a Power BI report that contains a paginated report visual

This is supported for all embedding scenarios now.

When embedding a paginated Report visual in a Power BI report with the embed for your customers method, you need a service principal and a multi-resource embed token, that contains the following:

- Power BI report ID
- Paginated report ID (of the report the visual is connected to)
- Dataset ID (of the Power BI report).

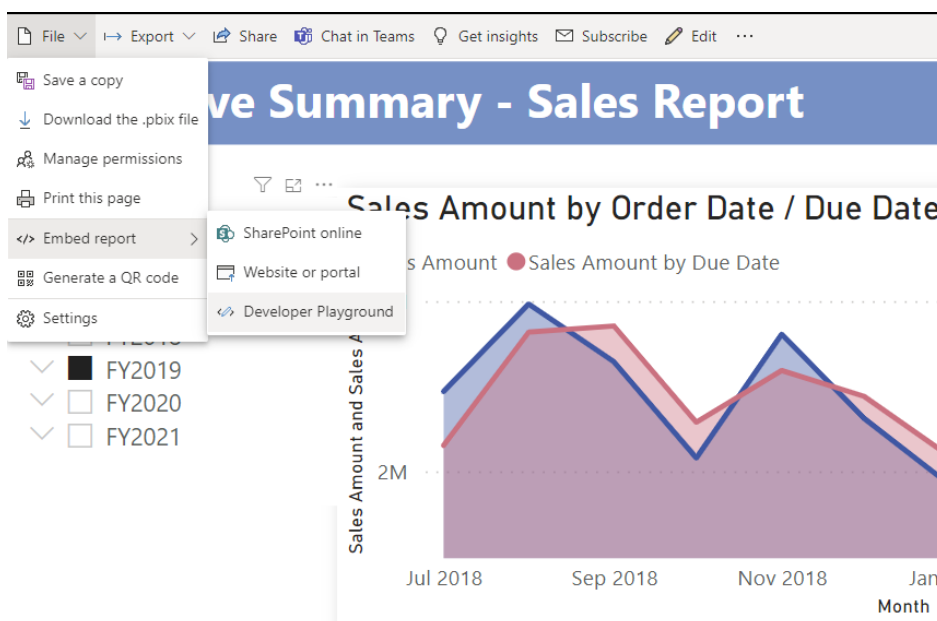
The service principal must have access to both reports (paginated and Power BI). Master user is not supported. Here is a request body example for the generate token call:

```
{
  "datasets": [
    {
      "id": "a5d577c7-0568-4180-a6d3-0f6cc0ca3df4"
    }
  ],
  "reports": [
    {
      "allowEdit": false,
      "id": "05024421-b4df-483c-a2ce-61202d0323ce"
    },
    {
      "id": "f8612306-f3a8-40e1-a448-d8e05992a007"
    }
  ]
}
```

Explore your report in the developer playground

If you're looking to embed an existing report in your website or portal, the latest update makes it easier for you to get started straight from the Power BI Service.

With your report open in the Power BI Service, you can now access the developer playground directly from the 'Embed report' menu under the 'File' menu in the Power BI Service.



Doing so will open the developer sandbox in the Power BI Embedded Analytics Playground with the same report loaded and ready to work with.

The developer sandbox is the place to explore all the features that Power BI Embedded has to offer and to get a feel of how your report would behave when embedded into your own app. All this may be achieved by simply dragging and dropping any of the APIs into the code editor and seeing

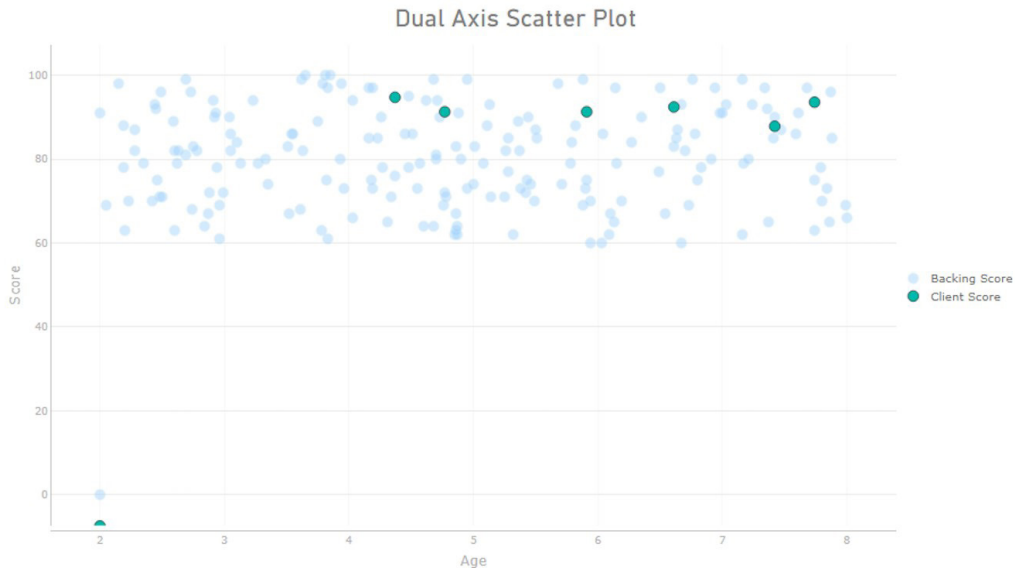
how they affect your report. You can check out the different APIs, save your code and explore the rest of the playground for more embedding information and inspiration.

Once you're ready to start embedding in your app, you can continue directly from the playground to set up your Power BI embedded environment.

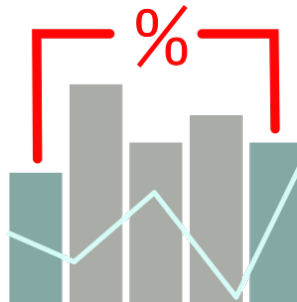
New visuals in AppSource

This month's latest visuals include:

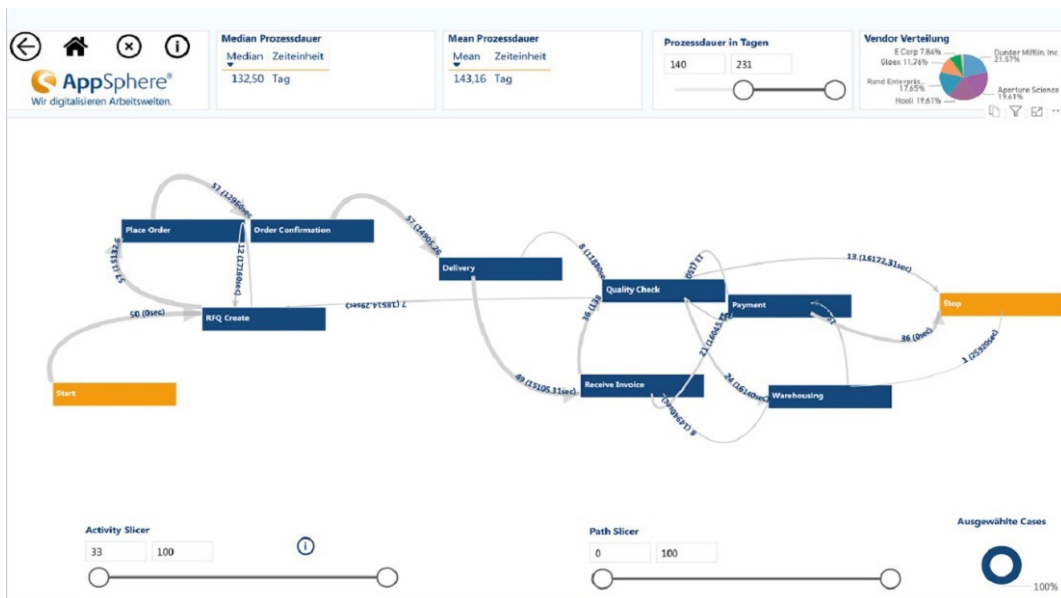
- Dual axis Scatter plot



- Growth Rate chart v1.1



- Process Mining with Power BI.

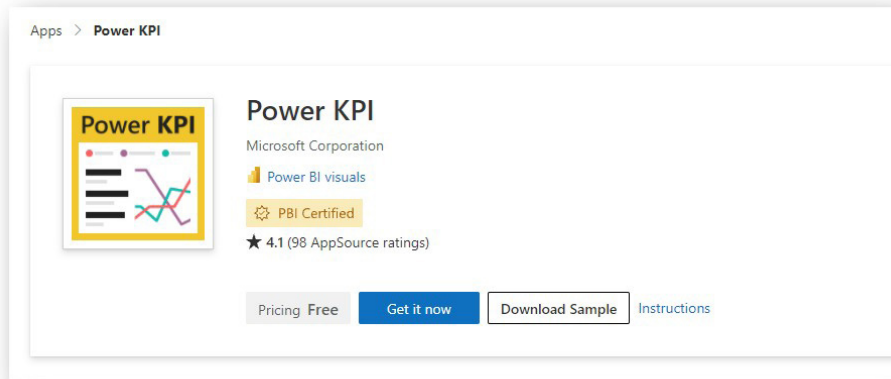


Find report sample files easily in AppSource

Report sample files are submitted by the visual's publisher together with the visual package during the submission process. These sample files help report creators understand how to use the visuals. They can see fields, datasets and tables that work with this visual in a real report sample. Many of these reports also include tips and tricks and further

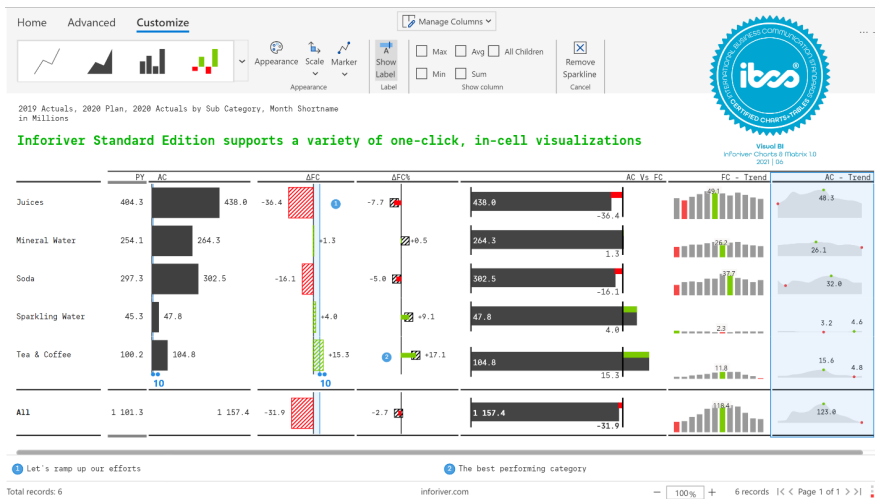
usage information from the publishers.

These files may be found in [AppSource.com](https://appsource.microsoft.com). Simply click on 'Download a sample' and the sample report will be downloaded to your computer. Duh!



Inforiver standard edition by Lumel

With Inforiver Standard Edition, you can create International Business Communication Standards (IBCS) compliant P&L / financial statements, variance reports and visualisations in a single click.



The standard edition comes with wide range of in-cell table / matrix visualisations with interactive menu-based navigation and one-click formatting options for end-user self-service, interactivity and collaboration.

Inforiver Standard Edition features include:

- 1-Click, in-cell customizable visualisations:
 - o variance charts, including absolute & % variances (for comparison)
 - o five types of bullet charts (for performance)
 - o Spark Line, Spark Column, Spark Area, Win-Loss and their variants (for trends)
 - o Stacked Chart, Waterfall (for distribution)
 - o Bar & Lolli Pop (for Magnitudes)
- 1-Click IBCS Reporting:
 - o Inforiver is Certified for (IBCS) and provides 1-Click template-based automated IBCS compliant report authoring experience
- Reporting:
 - o expand / collapse row and column hierarchies
 - o smart pagination
 - o automatic number formatting with unified scaling (IBCS)
 - o click-and-drag row and column reordering
 - o grand & subtotal positioning (top or bottom) as well as split totals
 - o insert / edit / reorder rows with robust formula engine
 - o automatic support for ragged hierarchies

- Collaboration:
 - o Cell level notes and annotations
 - o footnotes with no requirement for complex server / database requirements.

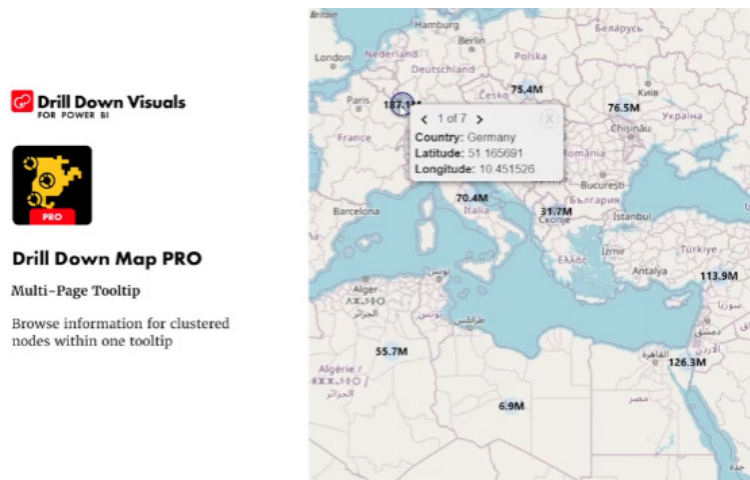
You can download Inforiver Standard Edition from AppSource.

Drill Down Map PRO by ZoomCharts

Drill Down Map PRO by ZoomCharts is a custom map visual for Power BI. It allows you to explore location-based data. You can explore data by country, city or use the lasso tool to create storable filters for custom regions.

Features include:

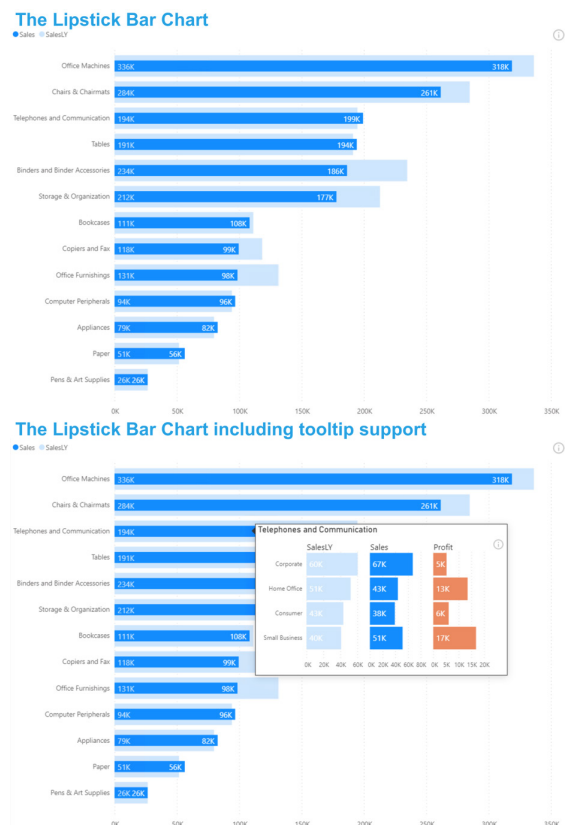
- **custom shape support:** provide your own custom shapes through KML and GeoJSON files; combine and stack up to 10 layers of shapes
- **lasso tool:** draw your own shapes on the fly and filter as you go
- **node clustering capabilities:** clusters may be formatted as donut charts of all categories within the cluster
- **various map layers let you choose between four different options:** namely Azure maps, Custom (OpenStreetMaps, Google, CartoDB etc.), Image (for example, floor plans) or None (just focus on shapes without background)
- **aura, image and custom label support.**



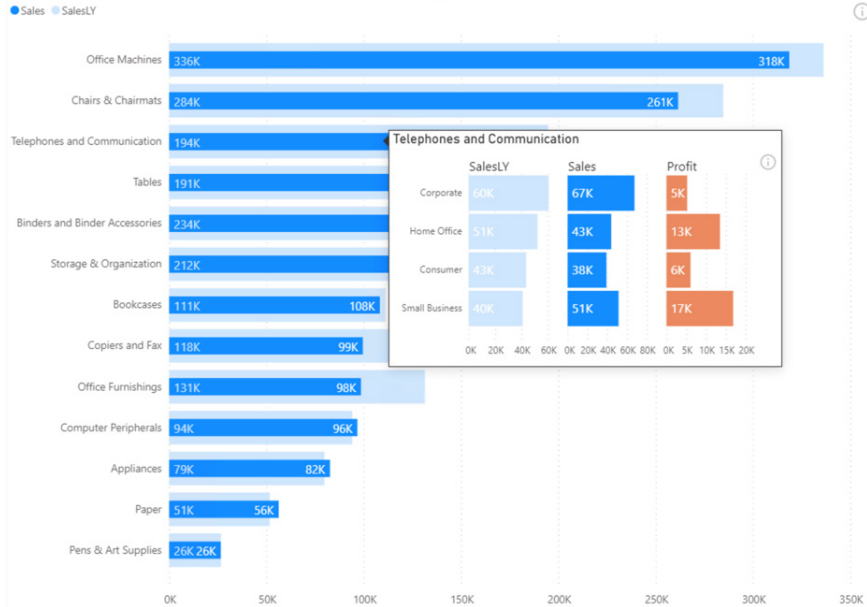
To evaluate the Drill Down Map PRO for free and obtain a 30 day trial period, download the visual from AppSource.

Lipstick Bar Chart by Nova Silva

This chart addresses the issue of comparing multiple series in a bar chart format. The Lipstick Bar Chart displays the two series on top of each other, and not side-by-side like the bar chart. This allows for an alternative comparison between the two series.



The Lipstick Bar Chart including tooltip support



All key features here work the same as those with the standard visuals. Again, it's in AppSource.

Charticulator (version 1.1.2)

Charticulator is tool for you to create custom charts for your reports using data modelled in Power BI. By representing data as glyphs and structuring the way they are repeated based on other data, you can create a diverse set of data visualizations along any permutation of mark and layout options.

The latest version of Charticulator supports a variety of new formatting features and user actions:

- support for the images table
- support for rotation of symbols
- support for tick format in numerical legend
- support for rectangular zoom for the glyph editor and chart editor
- support for axis offset in the plot segment
- introduction of a new categorical legend scale editor
- support for keyboard navigation for file view panels
- support for keyboard navigation for attributes panels.

There are also various bugfixes and UI improvements:

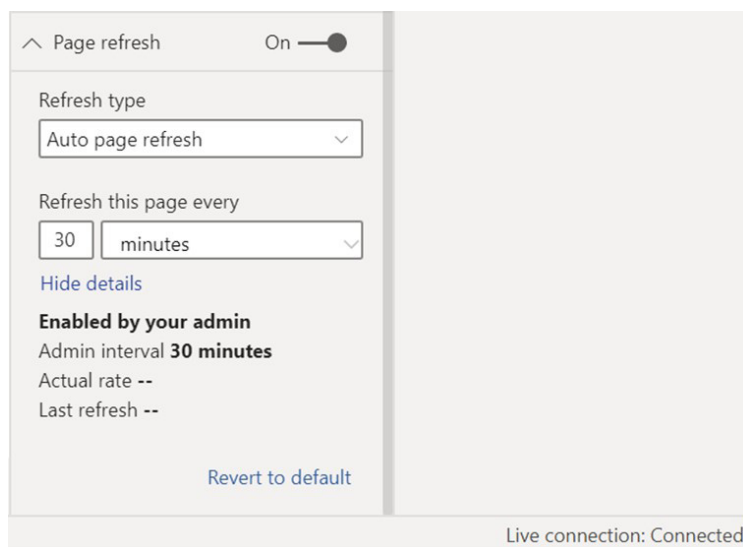
- fixed the save bug
- fixed the issue with items sorting order in categorical axis
- fixed the issue with scrolling when maximum is less than the minimum for a numerical range
- fixed the bug with the rounded corner for the rectangle mark
- fixed the bug with the automatic alignment of the Polar coordinate origin
- show the full image URL in a tooltip
- apply the Fluent UI for the 'Delete chart' modal dialog
- fixed the bug with scrolling and auto range configuration
- refined the image scale editor
- fixed the bug with the textbox opacity slider
- fixed the bug with the colour input.

Automatic Page Refresh now supports DirectQuery for Power BI datasets and Azure Analysis services

Last year, Microsoft made Automatic Page Refresh (APR) Generally Available. APR allows you to set the refresh interval of visuals in your report when using a DirectQuery source. There is now an incremental improvement to APR: it will now support DirectQuery for Power BI datasets and Azure Analysis services.

To enable APR for this feature, the process is the same as DirectQuery:

connect to a supported source, and head to the page format pane to find page refresh and then enable the toggle, select Auto page refresh, and define a duration for your refresh. The effective refresh interval for these datasets will depend upon the administrator interval settings where the dataset is hosted and, in the case of Power BI datasets, which type of workspace is hosting the dataset and its administrator interval settings for Automatic page refresh.

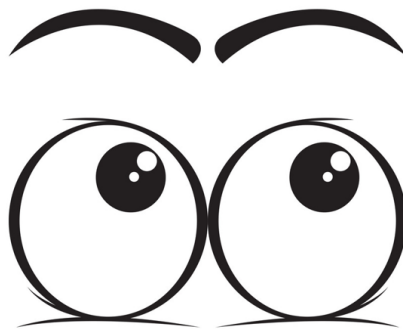


Please keep in mind the following limitations for APR in this scenario:

- the minimum refresh interval for DirectQuery for Power BI datasets and Azure Analysis services is 30 minutes in Power BI Desktop
- the effective refresh interval on Power BI Service is set to the maximum of the administrator settings for the local Models chain, Remote Model and Page Level Refresh Settings (*e.g.* if the DirectQuery for Power BI datasets and Azure Analysis services has a 30 minute refresh, Remote and Page are set to one second, then the effective refresh interval on Power BI Service will be 30 minutes)
- Power BI datasets are only supported if they are connected to a DirectQuery data source
- if your administrator has disabled Automatic Page Refresh in a Premium capacity, you will not be able to run autopage refresh
- change detection is not supported for this type of data source.

More next month, we're sure!

The A to Z of Excel Functions: HLOOKUP



Often we need to look up data in a table or list – and one such function many are familiar with is **HLOOKUP**. But do you realise it's very easy to make a mistake with this function.

HLOOKUP has the following syntax:

HLOOKUP(lookup_value, table_array, row_index_number, [range_lookup])

- **lookup_value**: what value do you want to look up?
- **table_array**: where is the lookup table?
- **row_index_number**: which row has the value you want returned?
- **[range_lookup]**: do you want an exact or an approximate match? This is optional and to begin with, I am going to ignore this argument exists.

VLOOKUP is similar, but works on a column, rather than a row, basis.

Example

HLOOKUP always looks for the **lookup_value** in the first row of a table (the **table_array**) and then returns a corresponding value so many rows below, determined by the **row_index_number**.

G25 =HLOOKUP(G23,G12:M18,7)

Example

Table

Lookup Value	1	2	3	4	5	6	7
Row 2	94	82	36	58	99	32	84
Row 3	40	77	75	28	96	48	47
Row 4	92	55	96	57	50	81	80
Row 5	61	47	45	51	27	39	35
Row 6	11	52	67	86	96	11	66
Row 7	76	96	29	84	45	26	44

HLOOKUP

Value 2

Result 96 =HLOOKUP(G23,G12:M18,7)

In this above example, the formula in cell G25 seeks the value 2 in the first row of the table G12:M18 and returns the corresponding value from the seventh row of the table (returning 96).

Pretty easy to understand; so far so good. So what goes wrong? Well, what happens if you add or remove a row from the table range?

Adding (inserting) a row gives us the wrong value:

G26 =HLOOKUP(G24,G12:M19,7)

Example

Table

Lookup Value	1	2	3	4	5	6	7
Row 2	94	82	36	58	99	32	84
Row 3	40	77	75	28	96	48	47
Row 4	92	55	96	57	50	81	80
Row 5	61	47	45	51	27	39	35
INSERTED							
Row 6	11	52	67	86	96	11	66
Row 7	76	96	29	84	45	26	44

HLOOKUP

Value 2

Result 52 =HLOOKUP(G24,G12:M19,7)

With a row inserted, the formula contains hard code (7) and therefore, the seventh row (row 18, not row 19) is still referenced, giving rise to the wrong value. Deleting a row instead is even worse:

G24 =HLOOKUP(G22,G12:M17,7)

Example

Table

Lookup Value	1	2	3	4	5	6	7
Row 2	94	82	36	58	99	32	84
Row 3	40	77	75	28	96	48	47
Row 4	92	55	96	57	50	81	80
Row 6	11	52	67	86	96	11	66
Row 7	76	96	29	84	45	26	44

HLOOKUP

Value 2

Result #REF! =HLOOKUP(G22,G12:M17,7)

Now there are only six rows so the formula returns #REF! Oops.

It is possible to make the row index number dynamic using the ROWS function (that's right, every ROWS has its HLOOKUP function):

G25 =HLOOKUP(G23,G12:M18,ROWS(F12:F18))

Example

Table

Lookup Value	1	2	3	4	5	6	7
Row 2	94	82	36	58	99	32	84
Row 3	40	77	75	28	96	48	47
Row 4	92	55	96	57	50	81	80
Row 5	61	47	45	51	27	39	35
Row 6	11	52	67	86	96	11	66
Row 7	76	96	29	84	45	26	44

HLOOKUP

Value 2

Result 96 =HLOOKUP(G23,G12:M18,ROWS(F12:F18))

ROWS(reference) counts the number of rows in the **reference**. Using the range **F12:F18**, this formula will now keep track of how many rows there are between the lookup row (12) and the result row (18). This will prevent the problems illustrated above.

But there are more issues. Consider duplicate values in the lookup row. With one duplicate, the following happens:

G25 =HLOOKUP(G23,G12:M18,ROWS(F12:F18))

Example

Table

Lookup Value	2	2	3	4	5	6	7
Row 2	94	82	36	58	99	32	84
Row 3	40	77	75	28	96	48	47
Row 4	92	55	96	57	50	81	80
Row 5	61	47	45	51	27	39	35
Row 6	11	52	67	86	96	11	66
Row 7	76	96	29	84	45	26	44

HLOOKUP

Value 2

Result 96 =HLOOKUP(G23,G12:M18,ROWS(F12:F18))

Here, the second value is returned, which might not be what is wanted. With two duplicates:

G25 =HLOOKUP(G23,G12:M18,ROWS(F12:F18))

Example

Table

Lookup Value	2	2	2	4	5	6	7
Row 2	94	82	36	58	99	32	84
Row 3	40	77	75	28	96	48	47
Row 4	92	55	96	57	50	81	80
Row 5	61	47	45	51	27	39	35
Row 6	11	52	67	86	96	11	66
Row 7	76	96	29	84	45	26	44

HLOOKUP

Value 2

Result 29 =HLOOKUP(G23,G12:M18,ROWS(F12:F18))

Ah, it looks like it might take the last occurrence. Testing this hypothesis with three duplicates:

G25 \times \checkmark f_x =HLOOKUP(G23,G12:M18,ROWS(F12:F18))

Lookup Value	2	2	2	2	5	6	7
Row 2	94	82	36	58	99	32	84
Row 3	40	77	75	28	96	48	47
Row 4	92	55	96	57	50	81	80
Row 5	61	47	45	51	27	39	35
Row 6	11	52	67	86	96	11	66
Row 7	76	96	29	84	45	26	44

HLOOKUP

Value 2

Result 84 =HLOOKUP(G23,G12:M18,ROWS(F12:F18))

Yes, there seems to be a pattern: **HLOOKUP** takes the last occurrence. Better make sure:

G25 \times \checkmark f_x =HLOOKUP(G23,G12:M18,ROWS(F12:F18))

Lookup Value	2	2	2	2	2	6	2
Row 2	94	82	36	58	99	32	84
Row 3	40	77	75	28	96	48	47
Row 4	92	55	96	57	50	81	80
Row 5	61	47	45	51	27	39	35
Row 6	11	52	67	86	96	11	66
Row 7	76	96	29	84	45	26	44

HLOOKUP

Value 2

Result 45 =HLOOKUP(G23,G12:M18,ROWS(F12:F18))

Rats. In this example, the value returned is the fifth of six. The problem is, there's no consistent logic and the formula and its result cannot be relied upon. It gets worse if we exclude duplicates but mix up the lookup row a little:

G25 \times \checkmark f_x =HLOOKUP(G23,G12:L18,ROWS(F12:F18))

Lookup Value	5	2	3	-4	1	5
Row 2	1	82	36	58	99	32
Row 3	40	77	75	28	96	48
Row 4	92	55	96	57	50	81
Row 5	61	47	45	51	27	39
Row 6	11	52	67	86	96	11
Row 7	76	96	29	84	45	26

HLOOKUP

Value 2

Result #N/A =HLOOKUP(G23,G12:L18,ROWS(F12:F18))

In this instance, **HLOOKUP** cannot even find the value 2!

So what's going on? The problem – and common modelling mistake – is that the fourth argument has been ignored:

HLOOKUP(lookup_value, table_array, row_index_number, [range_lookup])

[range_lookup] appears in square brackets, which means it is optional. It has two values:

- **TRUE:** this is the default setting if the argument is not specified. Here, **HLOOKUP** will seek an approximate match, looking for the largest value less than or equal to the value sought. There is a price to be paid though: the values in the first row must be in strict ascending order – this means that each value must be larger than the value before, so no duplicates.

This is useful when looking up postage rates for example where prices are given in categories of pounds and you have 2.7lb to post (say). It's worth noting though that this isn't the most common lookup when modelling.

- **FALSE:** this has to be specified. In this case, data can be any which way – including duplicates – and the result will be based upon the first occurrence of the value sought. If an exact match cannot be found, **HLOOKUP** will return the value **#N/A**.

And this is the problem highlighted by the above examples. The final argument was never specified so the lookup row data has to be in strict ascending order – and this premiss was continually breached.

The robust formula needs both **ROWS** and a fourth argument of **FALSE** to work as expected:

G25 =HLOOKUP(G23,G12:M18,ROWS(F12:F18),FALSE)

Lookup Value	6	2	#DIV/0!	2	2	-5	2
Row 2	94	82	36	58	99	32	84
Row 3	40	77	75	28	96	48	47
Row 4	92	55	96	57	50	81	80
Row 5	61	47	45	51	27	39	35
Row 6	11	52	67	86	96	11	66
Row 7	76	96	29	84	45	26	44

HLOOKUP

Value: 2

Result: 96

=HLOOKUP(G23,G12:M18,ROWS(F12:F18),FALSE)

This is a very common mistake in modelling. Using a fourth argument of **FALSE**, **HLOOKUP** will return the corresponding result for the first occurrence of the **lookup_value**, regardless of number of duplicates, errors or series order. If an approximate match is required, the data must be in strict ascending order.

HLOOKUP is not the simple, easy to use functions people think it is. In fact, it can never be used to return data for rows above.

Word to the Wise

VLOOKUP works like **HLOOKUP** but hunts out a value in the first column of a table and returns a value so many columns to the right of this reference. However, it has the same limitations and should be used just as carefully.

The A to Z of Excel Functions: HOUR



This function returns the hour of a time value (expecting a serial number). The hour is given as an integer, ranging from 0 (12:00 A.M.) to 23 (11:00 P.M.). The **HOUR** function employs the following syntax to operate:

HOUR(serial_number)

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

- serial_number: this is required and represents the time that contains the hour you want to find. Times may be entered as text strings within quotation marks (e.g. "6:45 PM"), as decimal numbers (for example, 0.78125, which represents 6:45 PM, or as results of other formulas or functions (for example, **TIMEVALUE("6:45 PM")**).

It should be further noted that:

- time values are a portion of a date value and are represented by a decimal number (e.g. 12:00 PM is represented as 0.5 because it is half of a day).

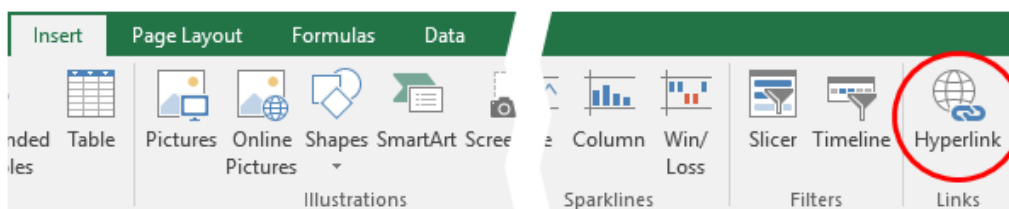
Please see our example below:

	A	B	C
1	Data		
2	0.75		
3	18/07/2020 07:45		
4	29/02/2020		
5			
6			
7	Formula	Description	Result
8	=HOUR(A2)	Returns 75% of 24 hours	18
9	=HOUR(A3)	Returns the hour portion of the date / time value	7
10	=HOUR(A4)	A date with no time portion specified is considered 12:00 AM (midnight) or zero (0) hours	0
11			

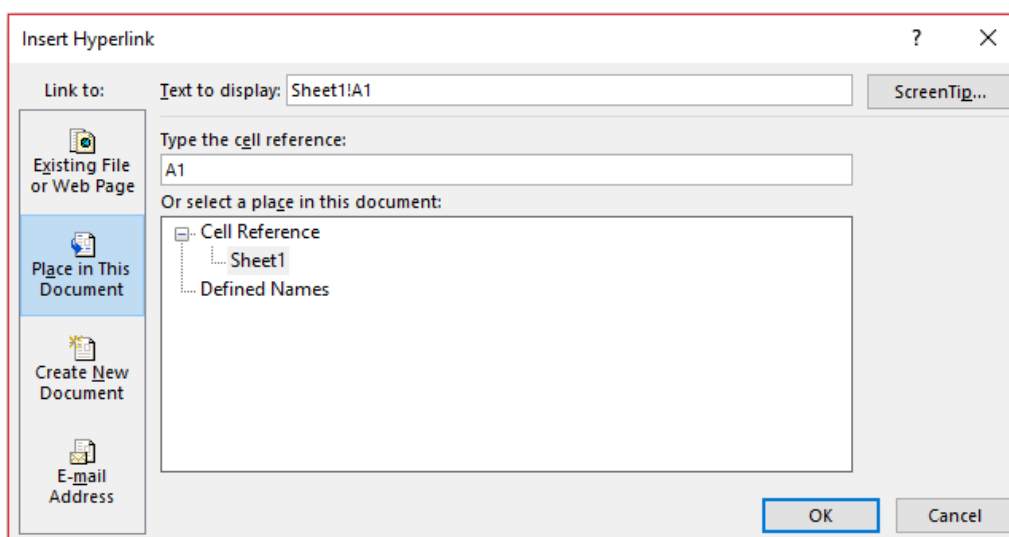
The A to Z of Excel Functions: HYPERLINK

We commonly use hyperlinks in our Excel files as they are a great way to move around a file. If you create a central worksheet with hyperlinks to all of the other worksheets, you are only ever two clicks away from anywhere else in the workbook. They make life very easy for end users and once you know how to construct them, they take mere seconds to insert.

The 'Insert Hyperlink' dialog box is fairly straightforward to use and readily accessed via one of two keyboard shortcuts, either **ALT + I + I** or **CTRL + K**. Alternatively, from the Ribbon, select the 'Insert' tab, click on 'Hyperlink' in the 'Links' group:



Hyperlinks may be used to link to a variety of places, but in this instance, let's focus on linking to elsewhere within the same workbook.



To create a hyperlink, first select the cell or range of cells that you wish to act as the hyperlink (*i.e.* clicking on any of these cells will activate the hyperlink). Then, open the Insert Hyperlink dialog box (above) and select 'Place in This Document' as the 'Link to:', which will change the appearance of the rest of the dialog box.

Insert the text for the hyperlink in the 'Text to display:' input box (clicking on the 'ScreenTip...' macro button will allow you to create an informative message in a message box when you hover over the hyperlink).

The next two input boxes, 'Type the cell reference:' and 'Or select a place in this document:', work in tandem – sort of:

- If you type a cell reference in the first input box without making a selection in the second input box, the hyperlink will link to the cell reference on the current (active) worksheet;
- If you type a cell reference in the first input box and select a worksheet reference in the second box, the hyperlink will link to the specified cell in the given worksheet. In my example above, this hyperlink will jump to **Sheet1** cell **A1**; or
- If you select a 'Defined Name' (*i.e.* a pre-defined range name) in the second input box, this will link to the cell(s) specified. This is the recommended option, where available, if you wish to link to cell(s) on another worksheet within the same workbook. This is because if the destination worksheet's sheet name were to be changed, the link would still work. I recommend that the range name should start with **HL_** for **H**yper **L**ink, to make it easier to sort through range names if necessary.

It should be noted that there is an Excel function. We tend not to use this as it is not so user-friendly. But let's go through it anyway. The **HYPERLINK** function creates a shortcut that jumps to another location in the current workbook or opens a document stored on a network server, an intranet or the Internet. When you click a cell that contains a **HYPERLINK** function, Excel jumps to the location listed, or opens the document you specified. It employs the following syntax:

HYPERLINK(link_location, [friendly_name])

The **HYPERLINK** function syntax has the following arguments:

- **link_location**: this is required and represents the path and file name to the document to be opened. The **link_location** can refer to a place in a document, such as a specific cell or named range in an Excel worksheet or workbook, or to a bookmark in a Microsoft Word document. The path can be to a file that is stored on a hard disk drive. The path can also be a universal naming convention (UNC) path on a server (in Microsoft Excel for Windows) or a Uniform Resource Locator (URL) path on the Internet or an intranet
- If it should be noted that on Excel for the web, the **HYPERLINK** function is valid for web addresses (URLs) only. The **link_location** can be a text string enclosed in quotation marks or a reference to a cell that contains the link as a text string
- if the jump specified in **link_location** does not exist or cannot be navigated, an error appears when you click the cell
- **friendly_name**: this argument is optional. This is the jump text or numeric value that is displayed in the cell. The **friendly_name** is displayed in blue and is underlined. If **friendly_name** is omitted, the cell displays the **link_location** as the jump text **friendly_name** may be a value, a text string, a name, or a cell that contains the jump text or value. If **friendly_name** returns an error value (for example, #VALUE!), the cell displays the error instead of the jump text.

It should be noted that in the Excel desktop application, to select a cell that contains a hyperlink without jumping to the hyperlink destination, click the cell and hold the mouse button until the pointer becomes a cross, then release the mouse button. In Excel for the web, select a cell by clicking it when the pointer is an arrow; jump to the hyperlink destination by clicking when the pointer is a pointing hand.

Please see our final example below:

	A	B
1	Example	Result
2	<code>=HYPERLINK("http://example.microsoft.com/report/budget report.xlsx", "Click for report")</code>	Opens a workbook saved at http://example.microsoft.com/report . The cell displays "Click for report" as its jump text
3	<code>=HYPERLINK("[http://example.microsoft.com/report/budget report.xlsx]Annual!F10", D1)</code>	Creates a hyperlink to cell F10 on the Annual worksheet in the workbook saved at http://example.microsoft.com/report . The cell on the worksheet that contains the hyperlink displays the contents of cell D1 as its jump text
4	<code>=HYPERLINK(\$A\$1)</code>	To quickly update all formulas in a worksheet that use a HYPERLINK function with the same arguments, you can place the link target in another cell on the same or another worksheet, and then use an absolute reference to that cell as the link_location in the HYPERLINK formulae. Changes that you make to the link target are immediately reflected in the HYPERLINK formulae
5		

More Excel Functions next month.

Beat the Boredom Suggested Solution

This month's challenge was one of our most complicated ones yet (not)! How do you enable the ability to filter on each column in a PivotTable? Here is an example:

Sum of Total Invoice	Column Labels				
Row Labels	Accessories	Bikes	Clothing	Grand Total	
Australia	\$153,253.45	\$9,781,515.48	\$77,637.33	\$10,012,406.26	
Canada	\$114,232.75	\$2,012,539.19	\$58,746.97	\$2,185,518.90	
France	\$70,064.64	\$2,821,701.23	\$29,873.95	\$2,921,639.83	
Germany	\$68,767.16	\$3,103,408.43	\$26,039.80	\$3,198,215.39	
United Kingdom	\$84,676.38	\$3,627,541.23	\$35,624.71	\$3,747,842.32	
United States	\$283,346.93	\$9,944,845.02	\$147,526.40	\$10,375,718.35	
Grand Total	\$774,341.31	\$31,291,550.59	\$375,449.16	\$32,441,341.06	

Sum of Total Invoice	Column Labels				
Row Labels	Accessories	Bikes	Clothing	Grand Total	
Canada			\$58,746.97	\$2,185,518.90	
France			\$29,873.95	\$2,921,639.83	
Germany			\$26,039.80	\$3,198,215.39	

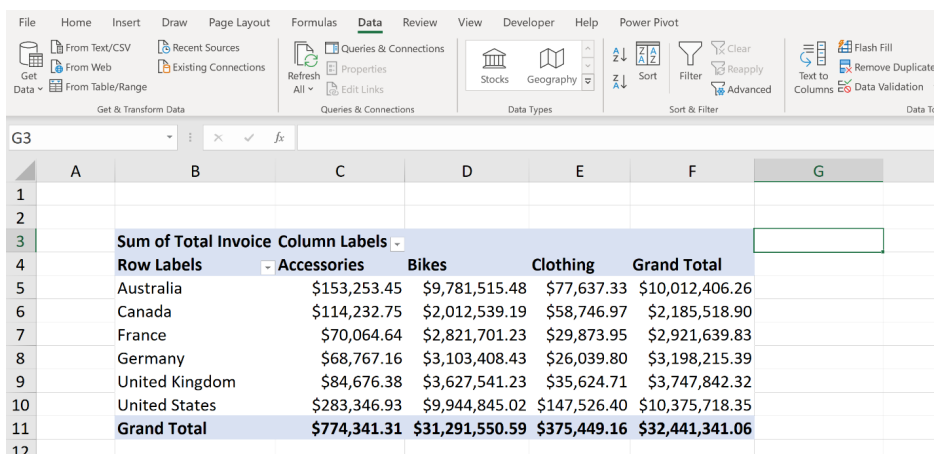
This had to be done with no VBA, no megaformulae, and not in Power BI, but in Excel.

You may have thought, hey, applying filters will be easy. I'll just highlight the columns in the PivotTable that I wish to apply the filters to, navigate to the Data tab on the Ribbon and click on the Filter option in the 'Sort & Filter' group.

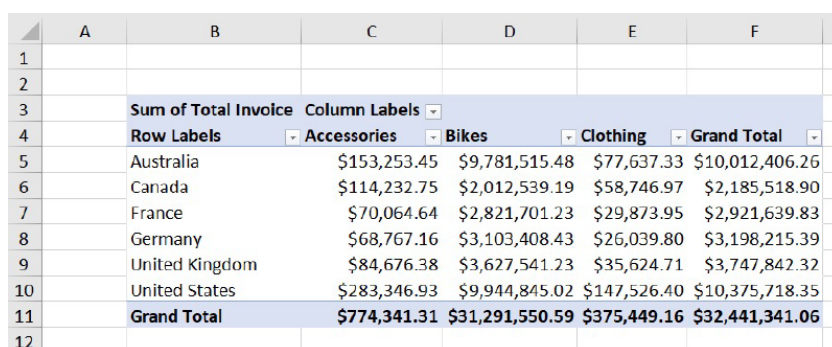
Sum of Total Invoice	Column Labels				
Row Labels	Accessories	Bikes	Clothing	Grand Total	
Australia	\$153,253.45	\$9,781,515.48	\$77,637.33	\$10,012,406.26	
Canada	\$114,232.75	\$2,012,539.19	\$58,746.97	\$2,185,518.90	
France	\$70,064.64	\$2,821,701.23	\$29,873.95	\$2,921,639.83	
Germany	\$68,767.16	\$3,103,408.43	\$26,039.80	\$3,198,215.39	
United Kingdom	\$84,676.38	\$3,627,541.23	\$35,624.71	\$3,747,842.32	
United States	\$283,346.93	\$9,944,845.02	\$147,526.40	\$10,375,718.35	
Grand Total	\$774,341.31	\$31,291,550.59	\$375,449.16	\$32,441,341.06	

But to your dismay, it is greyed out! So how on earth do we filter the PivotTable? If you thought the challenge was complicated, sit back, grab some scotch, or wine and get ready, this may be a long one. Or perhaps I am practicing poetic licence as this article is really short...

To enable filters on the PivotTable, we click the cell that is located one cell to the right of the top right-hand corner of the PivotTable:



Notice that the Filter option can now be toggled 'on'? In fact, this will work for any adjacent cell in column G. We click it and that's how we get our filters on each column in the PivotTable:



That was a tough one. Hopefully, you have not finished that bottle of alcohol yet! If you have, I suggest professional help – or else invite us round.

Until next time.

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	8 - 10 Dec 2021	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	15 Dec 2021	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	1 Day
Online (Australia)	Financial Modelling	16 - 17 Dec 2021	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	16 - 18 Feb 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	23 Feb 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	1 Day
Online (Australia)	Financial Modelling	24 - 25 Feb 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	2 Days
Online (Australia)	Excel Tips and Tricks	11 Apr 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	1 Day
Online (Australia)	Financial Modelling	12 - 13 Apr 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	10 - 12 May 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	17 May 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	1 Day

Location	Course	Date	Date	Duration	Duration
Online (Australia)	Financial Modelling	18 - 19 May 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	19 - 21 Jul 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	26 Jul 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	1 Day
Online (Australia)	Financial Modelling	27 - 28 Jul 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	2 Days
Online (Australia)	Excel Tips and Tricks	29 Aug 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	1 Day
Online (Australia)	Financial Modelling	30 - 31 Aug 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	28 -30 Sep 2022	09:00-17:00 AEST	(-1 day) 23:00-07:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	5 Oct 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	1 Day
Online (Australia)	Financial Modelling	6 - 7 Oct 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	9 - 11 Nov 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	16 Nov 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	1 Day
Online (Australia)	Financial Modelling	17 - 18 Nov 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	7 - 9 Dec 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	14 Dec 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	1 Day
Online (Australia)	Financial Modelling	15 - 16 Dec 2022	09:00-17:00 AEDT	(-1 day) 22:00-06:00 GMT	2 Days

Key Strokes

Each newsletter, we'd like to introduce you to useful keystrokes you may or may not be aware of. This year, we thought we'd revisit each function key in depth (there are 12 – one for each month of the year!). Given it's now December, let's look at the **F12** tips.

Keystroke	What it does
F12	Save as
CTRL + F12	Open
SHIFT + F12	Save
CTRL + ALT + F12	Thai dictionary
CTRL + SHIFT + F12	Print

Hey, it's not our fault so many do the same thing this month!

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