Sum Product

NEWSLETTER #102 - May 2021

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

I FORECAST another large newsletter this month…

With the A to Z of Excel Functions well and truly 'F'd' up, we head into the **FORECAST** family and use the excursion as an excuse to consider forecasting in Excel in general. I predict the usual ramble ahead!

As usual, that's not all. The Power BI updates continue to run amok, combined with another Beat the Boredom Challenge, plus Visual Basics, Power Pivot Principles, Power Query Pointers, Keyboard Shortcuts and the continuation of our new Charts and Dashboards series.

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

Until next month.

Liam Bastick, Managing Director, SumProduct

Manual vs. Automatic Forecasting in Excel

Given this month's A to Z of Excel Functions, it seems like a good time to revisit forecasting in Excel. And hey, let's face an ugly truth in the world of finance / accounting. It's the operational managers at the coalface who have the best understanding of likely demand and expected costs, yet many do not have the necessary tools or financial skills to forecast to the level that senior management requires. Similarly, most analysts do have these skills but may not be close enough to the front line to be able to understand operations sufficiently.

Operational managers and analysts need to work together to improve forecasting. That goes without saying and is never a problem in reality as the two sides work beautifully together, skipping off into the sunset hand in hand after a productive day in the office, understanding each other's needs and issues.

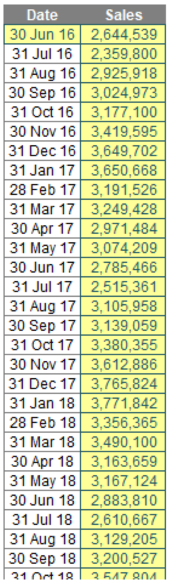
Yeah, right. If you believe that you're probably a Board Director…

Meanwhile, back on Planet Earth, often we find managers and analysts in a state of "co-operational flux" (a new term I have just invented). Managers sometimes feel accountants / analysts request forecasts from scratch: do they have either the time or skills to prepare a zero-based budget, whereas the analysts feel if they spend time preparing the base budget it is then torn apart by their operational counterpart. The art of war / budgeting: don't you just love it?

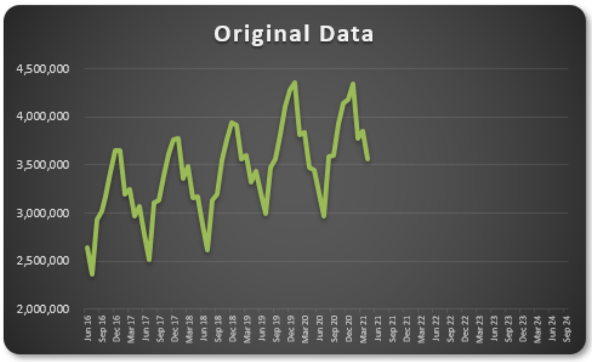
There is a need for objective forecasting. By this, I mean something that can be constructed simply such that if anyone follows the same process they will get the same figures. This needs to be a mechanical, objective process. This way, analysts may prepare this data in moments without feeling emotionally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

Wow, is that the longest preamble for an Excel article ever..?

I wanted to set the scene for having a simple mechanical approach for budgeting. Using the following example, let's take a look at ways Excel can do this for you. Imagine I have some historical data:



Date	Sales
30 Jun 16	2,644,539
31 Jul 16	2,359,800
31 Aug 16	2,925,918
30 Sep 16	3,024,973
31 Oct 16	3,177,100
30 Nov 16	3,419,595
31 Dec 16	3,649,702
31 Jan 17	3,650,668
28 Feb 17	3,191,526
31 Mar 17	3,249,428
30 Apr 17	2,971,484
31 May 17	3,074,209
30 Jun 17	2,785,466
31 Jul 17	2,515,361
31 Aug 17	3,105,958
30 Sep 17	3,139,059
31 Oct 17	3,380,355
30 Nov 17	3,612,886
31 Dec 17	3,765,824
31 Jan 18	3,771,842
28 Feb 18	3,356,365
31 Mar 18	3,490,100
30 Apr 18	3,163,659
31 May 18	3,167,124
30 Jun 18	2,883,810
31 Jul 18	2,610,667
31 Aug 18	3,129,205
30 Sep 18	3,200,527
31 Oct 18	3,547,804

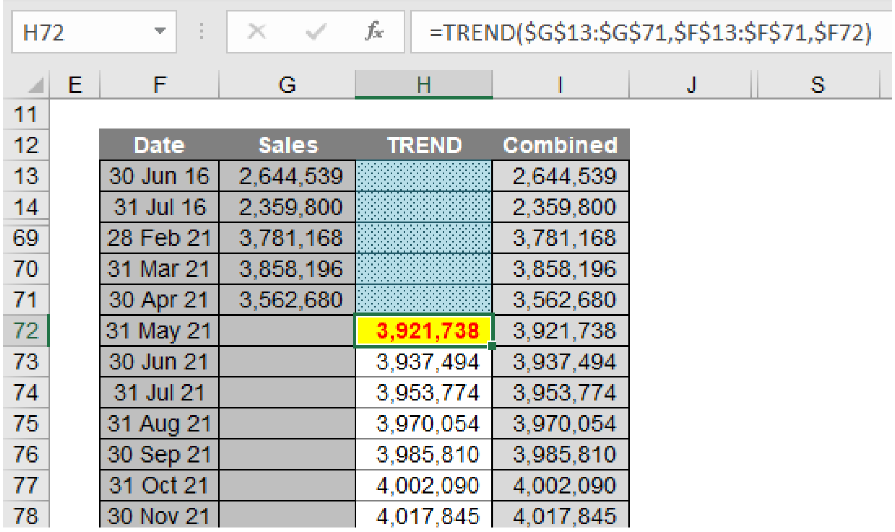


Now let's be honest, anyone who has historical data looking this perfect should be referred to the auditors, but hey, this is for illustration purposes. I have data from June 2016 to now (April 2021). I want to extrapolate it through sometime in 2024.

There are several functions that can help us here, with one of the simplest being **TREND**. **TREND(known_y's,known_x's,new_x's,[constant])** projects assuming that there is a relationship between two sets of variables **x** (independent variable – here, the dates) and **y** (dependent variable – the sales), through a formula **y = βx + c,** *i.e.* the equation of a straight line (**β** is the gradient of the line and **c** is the y-intercept).

Before you disregard linear regression, bear in mind many non-linear relationships can become linear ones by taking logarithms of the variables, for example. However, this won't always work.

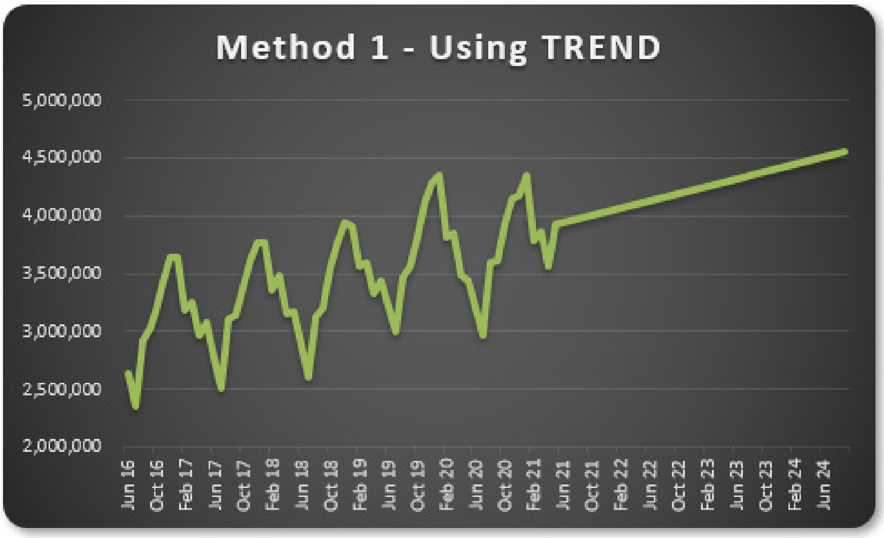
Here, time is our independent variable (**x**) and sales is our dependent variable (**y**). We only specify the **constant** if we want to force c in the equation (not common – it will usually be left blank). Similarly, it's preferable to leave constant blank in the **TREND** function. For example:



H72 | =TREND($G$13:$G$71,$F$13:$F$71,$F72)

	Date	Sales	TREND	Combined
13	30 Jun 16	2,644,539		2,644,539
14	31 Jul 16	2,359,800		2,359,800
69	28 Feb 21	3,781,168		3,781,168
70	31 Mar 21	3,858,196		3,858,196
71	30 Apr 21	3,562,680		3,562,680
72	31 May 21		3,921,738	3,921,738
73	30 Jun 21		3,937,494	3,937,494
74	31 Jul 21		3,953,774	3,953,774
75	31 Aug 21		3,970,054	3,970,054
76	30 Sep 21		3,985,810	3,985,810
77	31 Oct 21		4,002,090	4,002,090
78	30 Nov 21		4,017,845	4,017,845

Here, we can extrapolate the data using the **TREND** function *viz.*

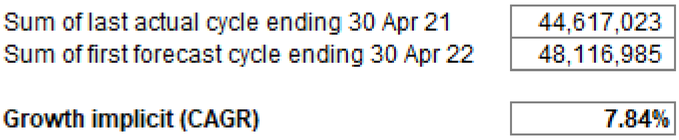


Ladies and gentleman, you may have heard of hockeystick projections; well, let me now present you with the swordfish. You extrapolate linearly, you get a *straight line*. Now who would like to present that projection to their senior management team?

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

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

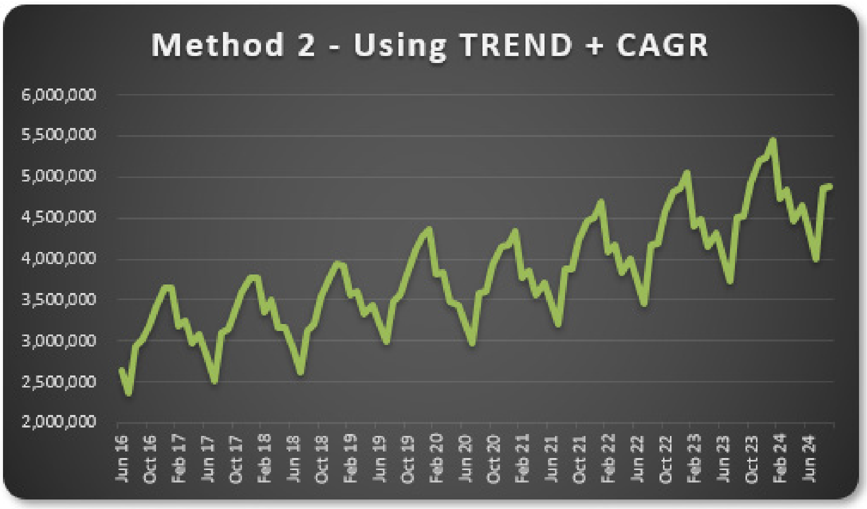
In my example, I am going to compare the sum of the sales over the 12 months ending 30 April 2021 with the forecast sales as calculated using **TREND** over the 12 months ending 30 April 2022:



Sum of last actual cycle ending 30 Apr 21	44,617,023
Sum of first forecast cycle ending 30 Apr 22	48,116,985
Growth implicit (CAGR)	**7.84%**

It is this percentage I will use to grow the forecasts (note the attached Excel file allows for different periodicities as long as the cycle remains constant).

I then grow each period's value by its corresponding value in the previous period by this percentage (7.84% here). This gives me a much better chart:



That looks *much* better. With practice, this approach doesn't take that long to prepare. Numbers may be varied from this forecast with the operational manager only having to explain these deviations. It makes life easier all round.

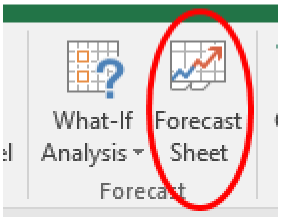
Once the method of assessing inferred growth rates based upon the **TREND** function have been agreed and what normalisations to historical data should be input, the process becomes much more straightforward. Of course, this method should be used for all forecast inputs separately and not just on their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

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

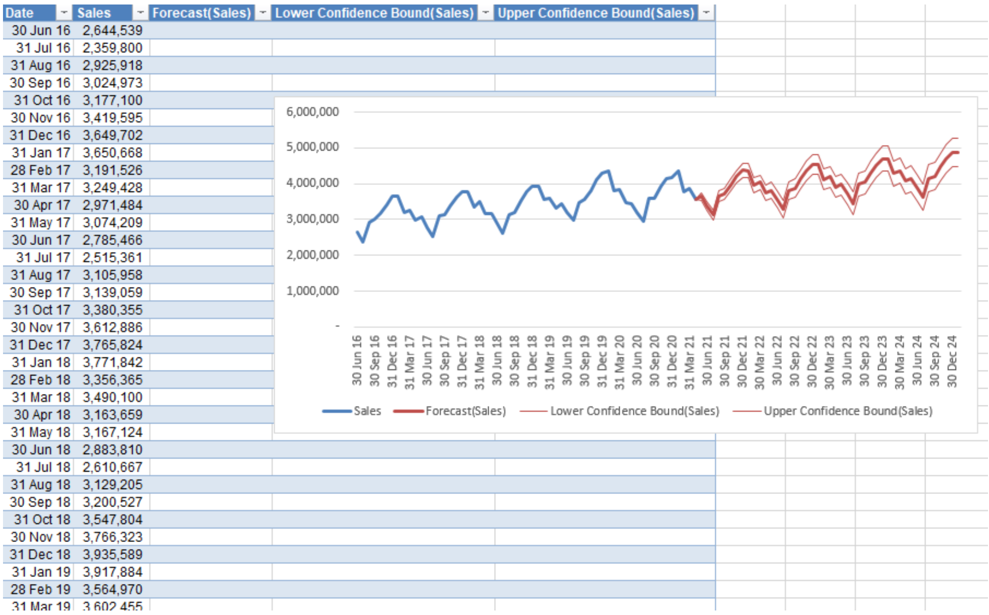
Exponential Triple Smoothing (ETS) sounds like a dairy process, but it actually uses the weighted mean of past values for forecasting. It's popular in statistics as it adjusts for seasonal variations in data, like in the example above. For those who really need to know, Excel uses a variation of the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

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

But don't worry about using these functions. Simply highlight the actual data and click on the 'Forecast Sheet' button in the 'Forecast' group of the 'Data' tab of the Ribbon (**ALT A + FC**):



All you need to do is specify the final forecast period at the prompt and that's it. It produces a raw data sheet, together with confidence intervals (to demonstrate potential spread in the forecast error), which looks something like this:



Date	Sales	Forecast(Sales)	Lower Confidence Bound(Sales)	Upper Confidence Bound(Sales)
30 Jun 16	2,644,539			
31 Jul 16	2,359,800			
31 Aug 16	2,925,918			
30 Sep 16	3,024,973			
31 Oct 16	3,177,100			
30 Nov 16	3,419,595			
31 Dec 16	3,649,702			
31 Jan 17	3,650,668			
28 Feb 17	3,191,526			
31 Mar 17	3,249,428			
30 Apr 17	2,971,484			
31 May 17	3,074,209			
30 Jun 17	2,785,466			
31 Jul 17	2,515,361			
31 Aug 17	3,105,958			
30 Sep 17	3,139,059			
31 Oct 17	3,380,355			
30 Nov 17	3,612,886			
31 Dec 17	3,765,824			
31 Jan 18	3,771,842			
28 Feb 18	3,356,365			
31 Mar 18	3,490,100			
30 Apr 18	3,163,659			
31 May 18	3,167,124			
30 Jun 18	2,883,810			
31 Jul 18	2,610,667			
31 Aug 18	3,129,205			
30 Sep 18	3,200,527			
31 Oct 18	3,547,804			
30 Nov 18	3,766,323			
31 Dec 18	3,935,589			
31 Jan 19	3,917,884			
28 Feb 19	3,564,970			
31 Mar 19	3,602,455			

Manual versus automatic?

- The “manual” method, using **TREND**, assumes some linear relationship (possibly at a derivative level) and requires some initial subjectivity regarding the normalisations of actual data and how to determine what growth rate to use over what duration. Once it has been agreed, it becomes a simple process both to understand and maintain.
- The “automatic” method using the Forecast Sheet works it all out at the press of a button, after normalisations to historical data have been made. However, it’s not all peaches and cream: it’s only available in Excel 2016, it’s quite “black box” to many and could you really explain to your line manager what Exponential Triple Smoothing is?

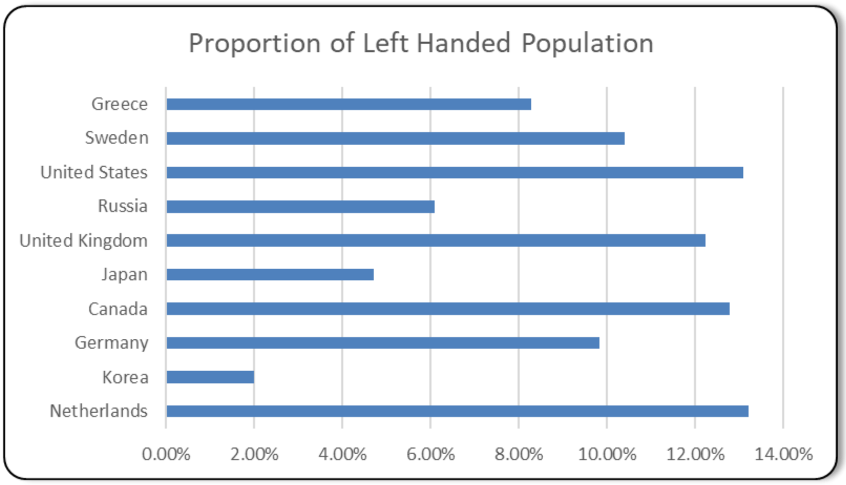
The jury remains out: whatever you decide, for your own sanity, I recommend an objective forecasting approach. And do check out this month’s A to Z of Excel Functions which discuss the family of **FORECAST** functions later in this newsletter…

Beat the Boredom Challenge

*With many of us currently “working from home” / quarantined, there are only so Zoom / Teams calls and virtual parties you can make before you reach your (data) limit. Perhaps they should measure data allowance in blood pressure millimetres of mercury (mmHg). To try and keep our readers engaged, we will continue to reproduce some of our popular **Final Friday Fix** challenges from yesteryear in this and upcoming newsletters. One suggested solution may be found later in this newsletter. Here’s this month’s…*

This time, let us return to charts. The Excel chart engine often does not always do what the people would like it to do and sometimes one is left scratching their heads as over simple formatting issues.

Consider the following bar chart, which shows some key statistics as follows:



We’d like the country labels to be left aligned, to coincide with the laterality of hands! What would be the best way to approach that?

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

Charts and Dashboards – a New Series

It’s time to chart our progress with an introductory series into the world of creating charts and dashboards in Excel. It’s time to create a chart.

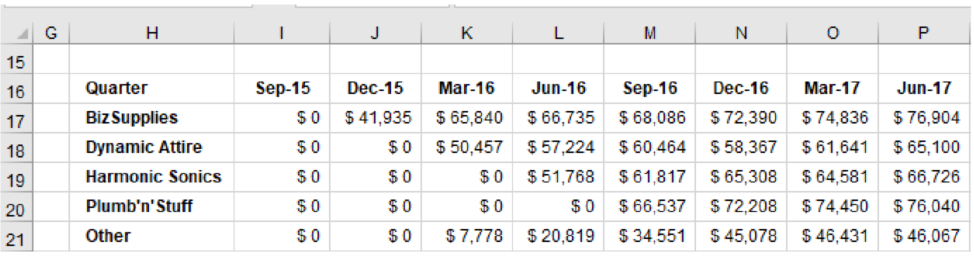
As they say, “a picture is worth a thousand words”. Well, there are always some people who will want to see the underlying data, but even the best analysts can still appreciate the value of communicating information using an image.

It is very important though when preparing charts that you do not influence the way the information is presented. The goal is to present the data visually for viewers to analyse and draw their own conclusions, not to present the data in such a way as to alter anyone’s perception of the data. As we proceed through this series, we will highlight tips about how to avoid effectively manipulating the presentation of the data.

To create a chart, the process is basically the same irrespective of the chart type you wish you use.

Step 1

To begin with, we need a table containing just the data I want to chart. We have a dataset of sales by customer groups by quarters as follows:

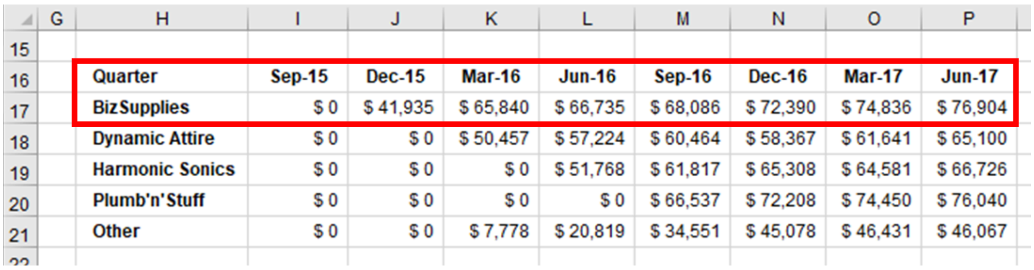


	G	H	I	J	K	L	M	N	O	P
15										
16		**Quarter**	**Sep-15**	**Dec-15**	**Mar-16**	**Jun-16**	**Sep-16**	**Dec-16**	**Mar-17**	**Jun-17**
17		**Biz Supplies**	$ 0	$ 41,935	$ 65,840	$ 66,735	$ 68,086	$ 72,390	$ 74,836	$ 76,904
18		**Dynamic Attire**	$ 0	$ 0	$ 50,457	$ 57,224	$ 60,464	$ 58,367	$ 61,641	$ 65,100
19		**Harmonic Sonics**	$ 0	$ 0	$ 0	$ 51,768	$ 61,817	$ 65,308	$ 64,581	$ 66,726
20		**Plumb'n'Stuff**	$ 0	$ 0	$ 0	$ 0	$ 66,537	$ 72,208	$ 74,450	$ 76,040
21		**Other**	$ 0	$ 0	$ 7,778	$ 20,819	$ 34,551	$ 45,078	$ 46,431	$ 46,067

Step 2

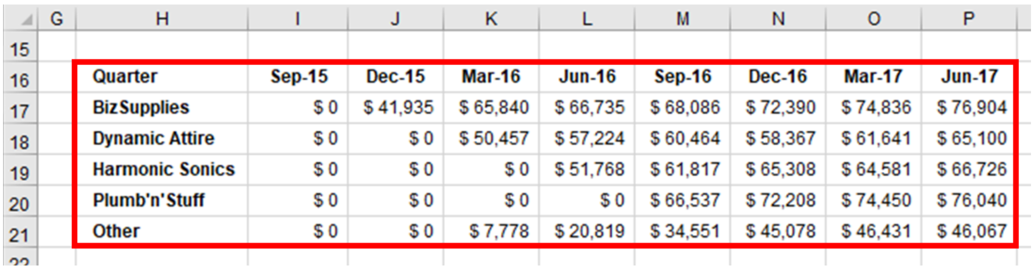
Once the data is ready, next we need to highlight the data to be included in the chart. We will highlight the headings above and to the left of our data, assuming we want these brought across to the chart. Most charts will allow for multiple data series to be reported in the one chart. The exception to this is the Pie Chart, where only one data series may be charted.

To highlight a single series, say the quarterly sales for **BizSupplies**, as per the example above, we would highlight from cells **H16** to **P17**:



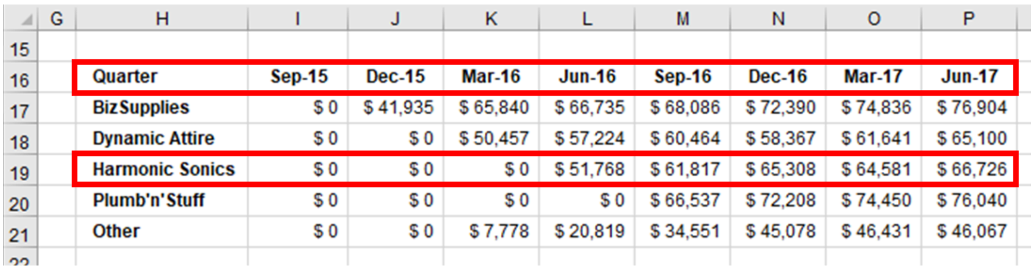
	G	H	I	J	K	L	M	N	O	P
15										
16		Quarter	Sep-15	Dec-15	Mar-16	Jun-16	Sep-16	Dec-16	Mar-17	Jun-17
17		BizSupplies	$ 0	$ 41,935	$ 65,840	$ 66,735	$ 68,086	$ 72,390	$ 74,836	$ 76,904
18		Dynamic Attire	$ 0	$ 0	$ 50,457	$ 57,224	$ 60,464	$ 58,367	$ 61,641	$ 65,100
19		Harmonic Sonics	$ 0	$ 0	$ 0	$ 51,768	$ 61,817	$ 65,308	$ 64,581	$ 66,726
20		Plumb'n'Stuff	$ 0	$ 0	$ 0	$ 0	$ 66,537	$ 72,208	$ 74,450	$ 76,040
21		Other	$ 0	$ 0	$ 7,778	$ 20,819	$ 34,551	$ 45,078	$ 46,431	$ 46,067

To highlight multiple series, say the quarterly sales for all clients, you would highlight from cells **H16** to **P21**:



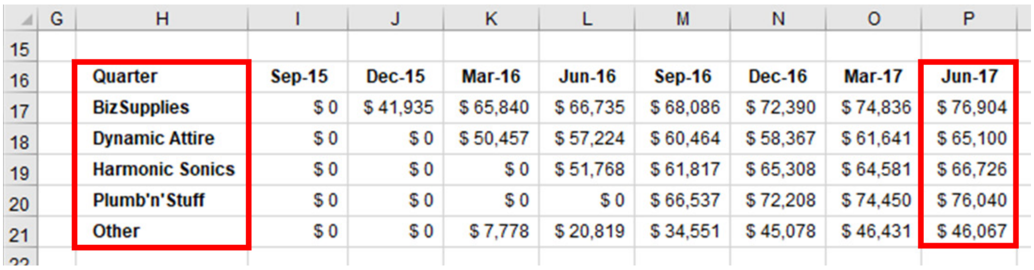
	G	H	I	J	K	L	M	N	O	P
15										
16		Quarter	Sep-15	Dec-15	Mar-16	Jun-16	Sep-16	Dec-16	Mar-17	Jun-17
17		BizSupplies	$ 0	$ 41,935	$ 65,840	$ 66,735	$ 68,086	$ 72,390	$ 74,836	$ 76,904
18		Dynamic Attire	$ 0	$ 0	$ 50,457	$ 57,224	$ 60,464	$ 58,367	$ 61,641	$ 65,100
19		Harmonic Sonics	$ 0	$ 0	$ 0	$ 51,768	$ 61,817	$ 65,308	$ 64,581	$ 66,726
20		Plumb'n'Stuff	$ 0	$ 0	$ 0	$ 0	$ 66,537	$ 72,208	$ 74,450	$ 76,040
21		Other	$ 0	$ 0	$ 7,778	$ 20,819	$ 34,551	$ 45,078	$ 46,431	$ 46,067

Please note that the data series do not have to be in a single area. For example, if you want to chart the quarterly sales for Harmonic Sonics, you may highlight the headings and the data series separately using the **CTRL** key. To achieve this, we would highlight cells **H16:P16**, then hold down the **CTRL** key and highlight **H19:P19**, and then release the **CTRL** key:



	G	H	I	J	K	L	M	N	O	P
15										
16		Quarter	Sep-15	Dec-15	Mar-16	Jun-16	Sep-16	Dec-16	Mar-17	Jun-17
17		BizSupplies	$ 0	$ 41,935	$ 65,840	$ 66,735	$ 68,086	$ 72,390	$ 74,836	$ 76,904
18		Dynamic Attire	$ 0	$ 0	$ 50,457	$ 57,224	$ 60,464	$ 58,367	$ 61,641	$ 65,100
19		Harmonic Sonics	$ 0	$ 0	$ 0	$ 51,768	$ 61,817	$ 65,308	$ 64,581	$ 66,726
20		Plumb'n'Stuff	$ 0	$ 0	$ 0	$ 0	$ 66,537	$ 72,208	$ 74,450	$ 76,040
21		Other	$ 0	$ 0	$ 7,778	$ 20,819	$ 34,551	$ 45,078	$ 46,431	$ 46,067

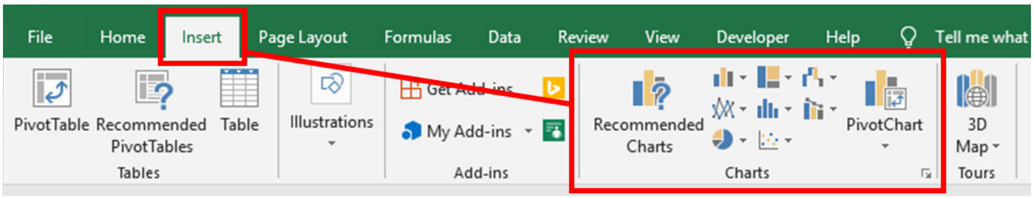
It is also important to note that data series do not have to be in rows. They may be in columns as well. Let's say you want to chart the quarterly income per customer for the June 2017 quarter. Here, you would highlight cells **H16:H21**, hold the **CTRL** key, then highlight **P16:P21**, and then release the **CTRL** key:



	G	H	I	J	K	L	M	N	O	P
15										
16		Quarter	Sep-15	Dec-15	Mar-16	Jun-16	Sep-16	Dec-16	Mar-17	Jun-17
17		BizSupplies	$ 0	$ 41,935	$ 65,840	$ 66,735	$ 68,086	$ 72,390	$ 74,836	$ 76,904
18		Dynamic Attire	$ 0	$ 0	$ 50,457	$ 57,224	$ 60,464	$ 58,367	$ 61,641	$ 65,100
19		Harmonic Sonics	$ 0	$ 0	$ 0	$ 51,768	$ 61,817	$ 65,308	$ 64,581	$ 66,726
20		Plumb'n'Stuff	$ 0	$ 0	$ 0	$ 0	$ 66,537	$ 72,208	$ 74,450	$ 76,040
21		Other	$ 0	$ 0	$ 7,778	$ 20,819	$ 34,551	$ 45,078	$ 46,431	$ 46,067

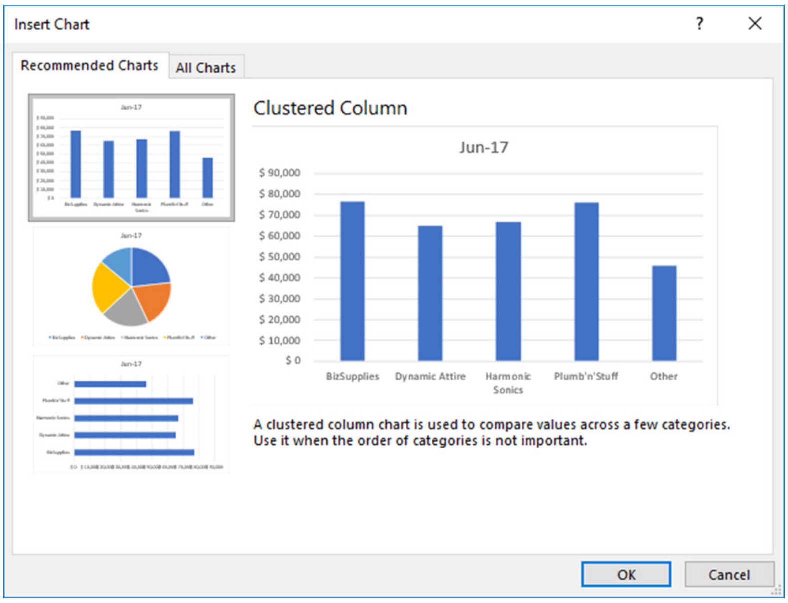
Step 3

With the data selected, we can now create a chart. In Excel, go to the Insert tab on the Ribbon. In the Charts group, we are presented with a number of options:



- if you click on 'Recommended Charts', Excel will analyse the data and provide you with a few options that it considers are appropriate. From this window, you may also click on the 'All Charts' tab at the top and see all the different chart types and all the variations available
- in the centre of the 'Charts' group, there are a series of buttons / icons representing the various chart types ready for you to select and use. Each has a drop-down arrow to show which charts have been grouped under each button and a subset of the variations available for each chart type
- the 'PivotChart' button links to 'PivotTables'. For the purposes of explaining charting, we will skip PivotCharts here (for the time being anyway!)
- finally, there is a little arrow in the bottom right corner of the Charts group. Clicking on this arrow will simply take me to the 'Recommended Charts' area.

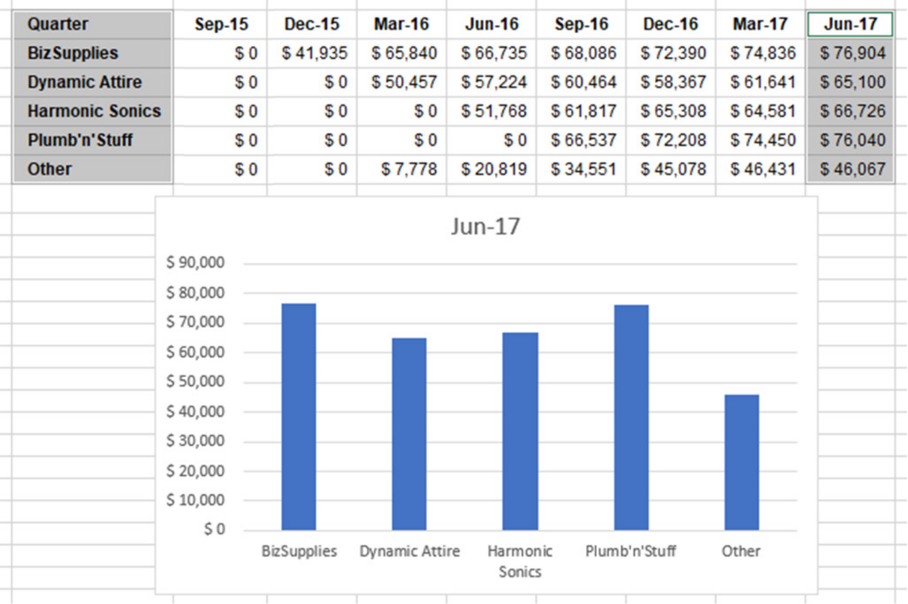
Let's click on 'Recommended Charts'. Using the available data where we have the June 2017 quarter data series in a column, Excel has provided the following 'Recommended Charts':



A clustered column chart is used to compare values across a few categories. Use it when the order of categories is not important.

We may choose one of these charts or alternatively click on the 'All Charts' tab at the top of this window and use any of the other chart types available.

Once you have the chart you want, simply click OK to accept your choice and Excel will embed the chart onto the spreadsheet where the data resides.



Quarter	Sep-15	Dec-15	Mar-16	Jun-16	Sep-16	Dec-16	Mar-17	Jun-17
BizSupplies	$ 0	$ 41,935	$ 65,840	$ 66,735	$ 68,086	$ 72,390	$ 74,836	$ 76,904
Dynamic Attire	$ 0	$ 0	$ 50,457	$ 57,224	$ 60,464	$ 58,367	$ 61,641	$ 65,100
Harmonic Sonics	$ 0	$ 0	$ 0	$ 51,768	$ 61,817	$ 65,308	$ 64,581	$ 66,726
Plumb'n'Stuff	$ 0	$ 0	$ 0	$ 0	$ 66,537	$ 72,208	$ 74,450	$ 76,040
Other	$ 0	$ 0	$ 7,778	$ 20,819	$ 34,551	$ 45,078	$ 46,431	$ 46,067

We'll continue next month…

Visual Basics

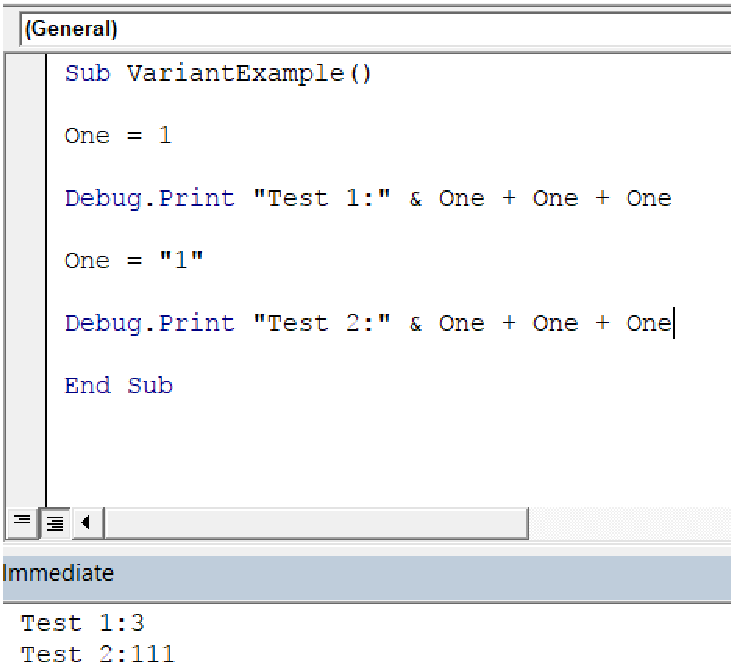
We thought we'd run an elementary series going through the rudiments of Visual Basic for Applications (VBA) as a springboard for newer users. This month, having previously looked at declaring variables, let's take an in depth look at good coding practice to and how it helps write better VBA code.

Efficiency

VBA data types determine the way in which data is stored in the computer's memory. It allocates the memory before the procedure is run instead of generating it on the fly. By default, when an undeclared variable is called upon, VBA assigns it as the Variant, which is the largest of the data types stored as 16 bytes. This can accumulate and take up a lot of memory and cause the execution to become sluggish. Think of it as a "Friday night variable".

Robustness

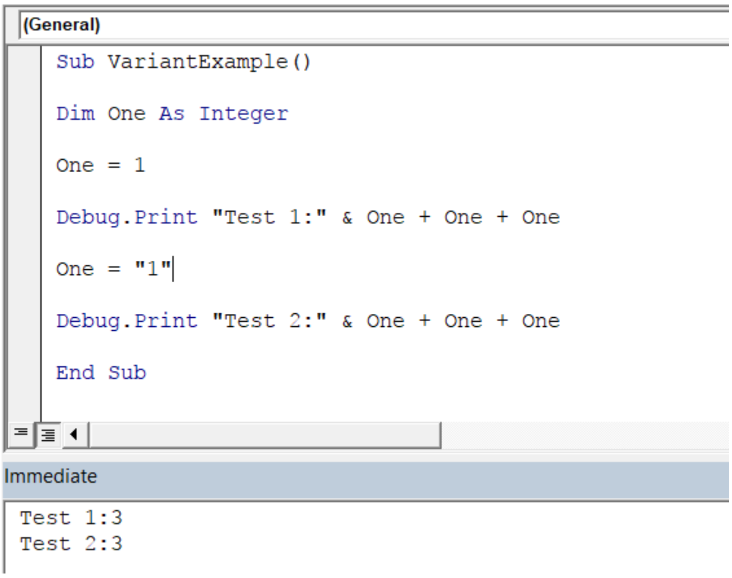
The problem with allowing the variable to be a Variant is that as the compiler tries to figure out what action you want to perform on it. It can cause unexpected results if you are not careful. Let's look at the following example:



```
Sub VariantExample()

One = 1

Debug.Print "Test 1:" & One + One + One

One = "1"

Debug.Print "Test 2:" & One + One + One

End Sub
```

```
Test 1:3
Test 2:111
```

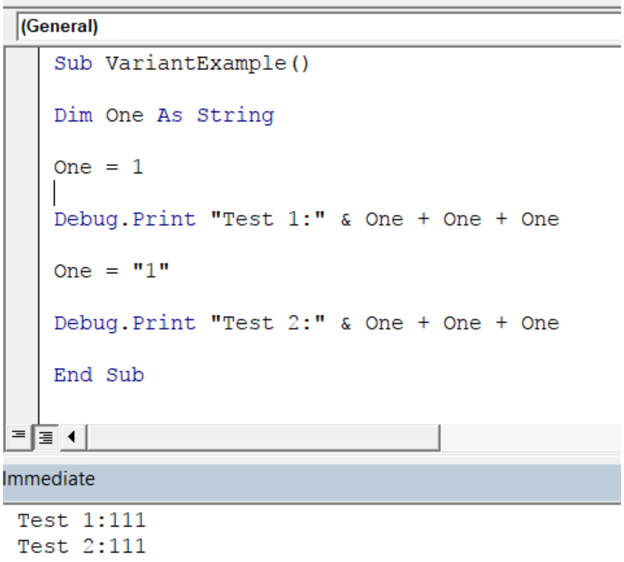
The variable 'One' in the first instance is used as a number and in the second instance as a string. However, as the type was not explicitly declared, VBA allowed it to be transformed mid routine. As a consequence, when it was called upon the second time, it was converted to string operation resulting in "111" instead of adding to three [3]. By declaring the variable as an Integer or Long (Numeric data types), VBA would have recognised the correct expected result using addition as a summing operator or alerted us upon compiling that there was a mistake in the code.



```
Sub VariantExample()

Dim One As Integer

One = 1

Debug.Print "Test 1:" & One + One + One

One = "1"

Debug.Print "Test 2:" & One + One + One

End Sub
```

```
Test 1:3
Test 2:3
```

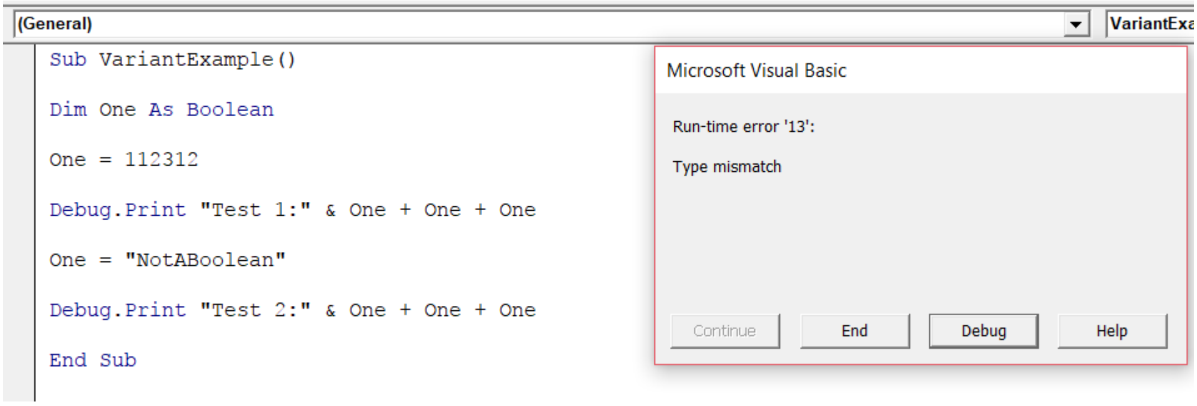
When the variable is declared to be an Integer explicitly, the compiler recognises that the quotation marks are superfluous and converts our number string to an actual number, which is what was actually expected and vice versa if declared as String:



```
Sub VariantExample()

Dim One As String

One = 1

Debug.Print "Test 1:" & One + One + One

One = "1"

Debug.Print "Test 2:" & One + One + One

End Sub
```

```
Test 1:111
Test 2:111
```

Now if it had been declared as a Boolean for an example, the compiler would give an error alert saying that the incorrect type has been used.



```
Sub VariantExample()

Dim One As Boolean

One = 112312

Debug.Print "Test 1:" & One + One + One

One = "NotABoolean"

Debug.Print "Test 2:" & One + One + One

End Sub
```

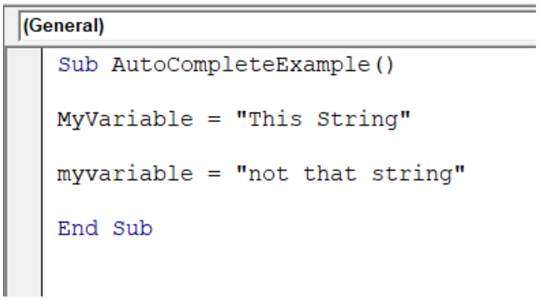
Microsoft Visual Basic

Run-time error '13':

Type mismatch

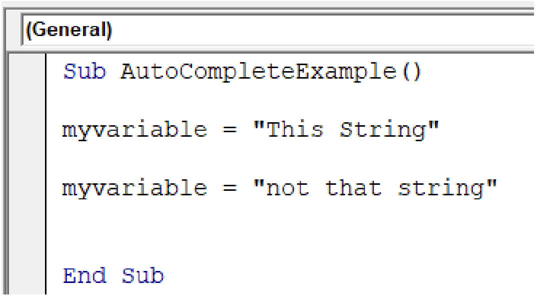
Consistency

Now the VBA editor does some weird things when variable is not declared. Let's reuse the variable whilst typing:



```
Sub AutoCompleteExample()

MyVariable = "This String"

myvariable = "not that string"

End Sub
```

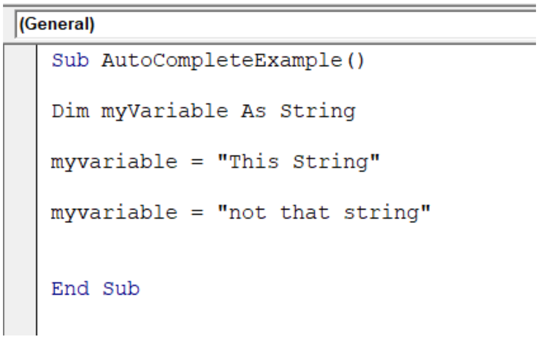
However, when **ENTER** is pressed right after, this happens:



```
Sub AutoCompleteExample()

myvariable = "This String"

myvariable = "not that string"


End Sub
```

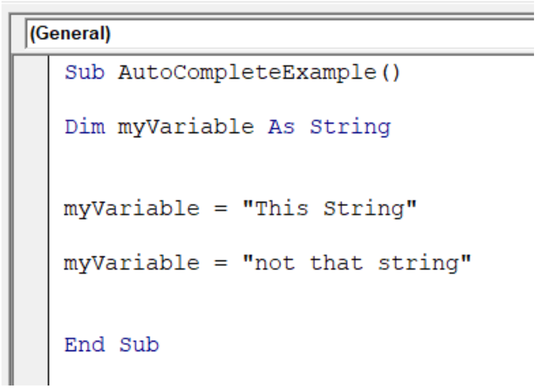
The capitalisation of the variable has changed! If the variable is not declared, then it will take the capitalisation properties of your most recent entry as opposed to the original. By declaring the variable with the capitalisation structure to be maintained, the editor autocorrects each instance to capitalisation to the declaration style, not the most recent.

After typing the declaration string changes all the code from this:



```
Sub AutoCompleteExample()

Dim myVariable As String

myvariable = "This String"

myvariable = "not that string"


End Sub
```

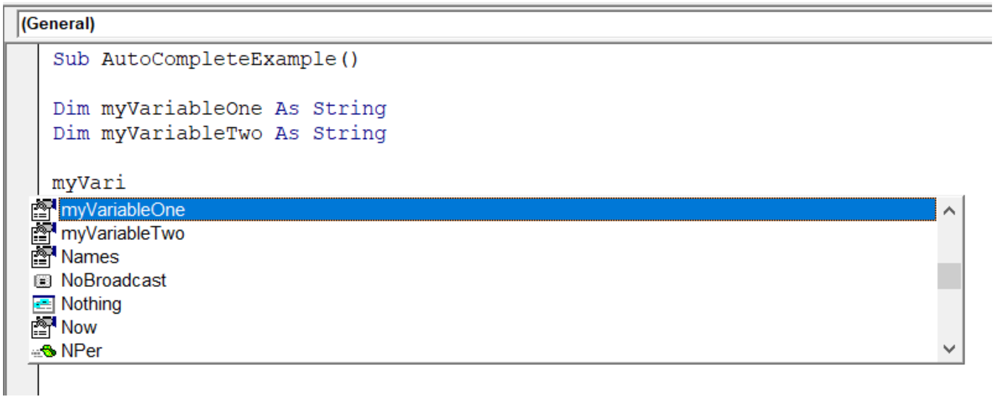
to this:



```
Sub AutoCompleteExample()

Dim myVariable As String


myVariable = "This String"

myVariable = "not that string"


End Sub
```

A nice thing about declaring a variable is that VBA editor's AutoComplete function can be utilised. By typing the first three characters (**'myV'**) and then pressing **CTRL + SPACE** it will autocomplete to the variable matching those letters (**'myVariable'**).

However, if two similarly named variables are available, a handy dialog will pop up allowing the coder to see what names are available for use:



```
Sub AutoCompleteExample()

Dim myVariableOne As String
Dim myVariableTwo As String

myVari
```

Declaring variables in VBA is always good coding practice. Not only is the code more efficient to run, it allows the programmer to use consistent form, check against errors and mitigate potentially rogue operations, so declare everything!

More next time.

Power Pivot Principles

We continue our series on the Excel COM add-in, Power Pivot. This month, we consider another way to use filters in Power Pivot.

Previously, we have discussed how to use filters in a PivotTable created with Power Pivot. This month, we will discuss how to incorporate filters directly into the measures we create in Power Pivot.

We begin with creating a new table with the following values:

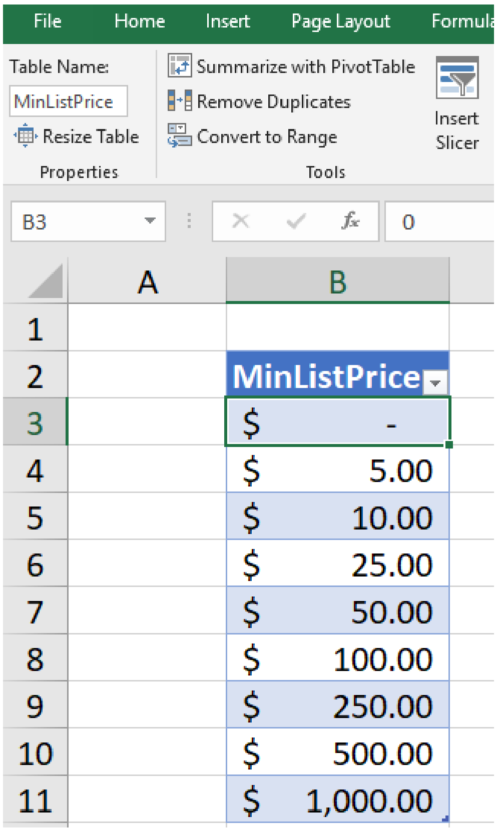


Table Name: MinListPrice

B3 = 0

	A	B
1		
2		MinListPrice
3		$ -
4		$ 5.00
5		$ 10.00
6		$ 25.00
7		$ 50.00
8		$ 100.00
9		$ 250.00
10		$ 500.00
11		$ 1,000.00

Let’s name the table **MinListPrice**. We will then add this table to our Data Model. After adding it to our Data Model we can create a new measure called **MinAmount** with the following formula:

=Min(MinListPrice[MinListPrice])

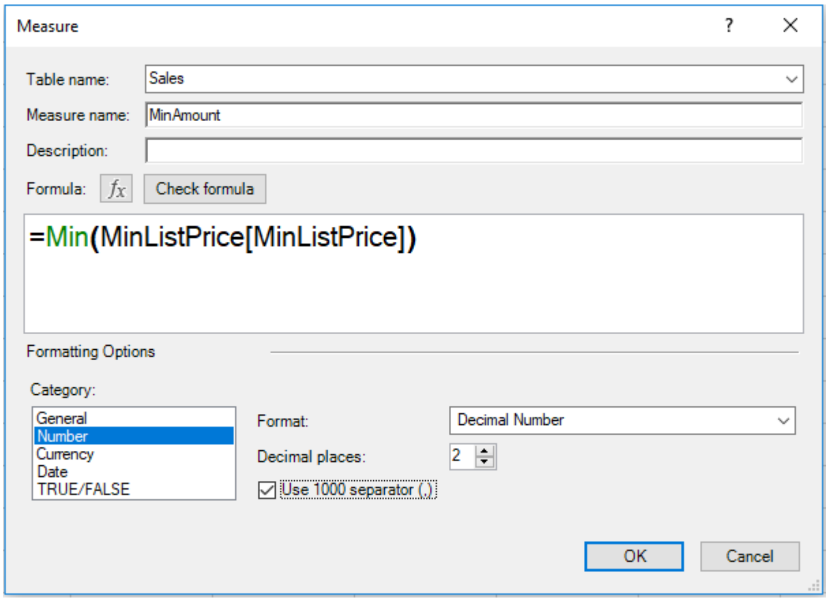


Table name: Sales

Measure name: MinAmount

Description:

Formula:

```
=Min(MinListPrice[MinListPrice])
```

Category: General / **Number** / Currency / Date / TRUE/FALSE

Format: Decimal Number

Decimal places: 2

☑ Use 1000 separator (,)

Our next move is to create another measure called **Sales Above Selected Unit List Price**, where the formula should be:

=CALCULATE([Sales],FILTER (Sales[UnitPrice]>=[MinAmount]))

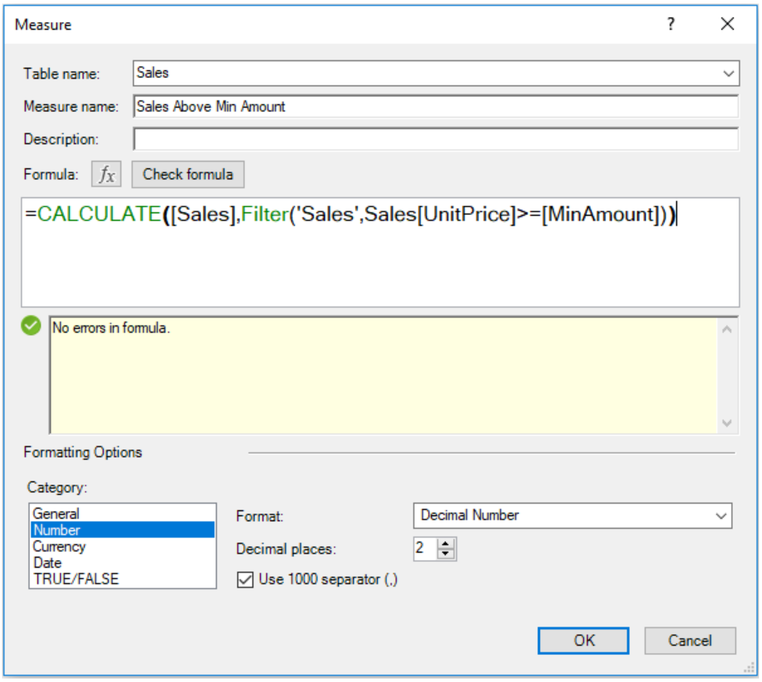


Table name: Sales

Measure name: Sales Above Min Amount

Description:

Formula:

```
=CALCULATE([Sales],Filter('Sales',Sales[UnitPrice]>=[MinAmount]))
```

No errors in formula.

Category: General, Number, Currency, Date, TRUE/FALSE

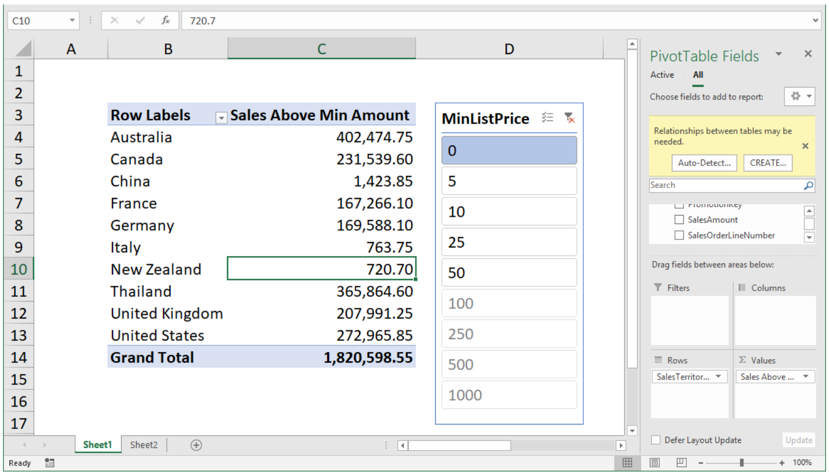
Format: Decimal Number

Decimal places: 2

☑ Use 1000 separator (,)

Note that this formula wouldn't work without using the **FILTER** function. This is because the calculation has to be performed on a record by record basis, rather than in aggregate, as **CALCULATE** usually works. This requires you to filter records, *i.e.* use the **FILTER** function.

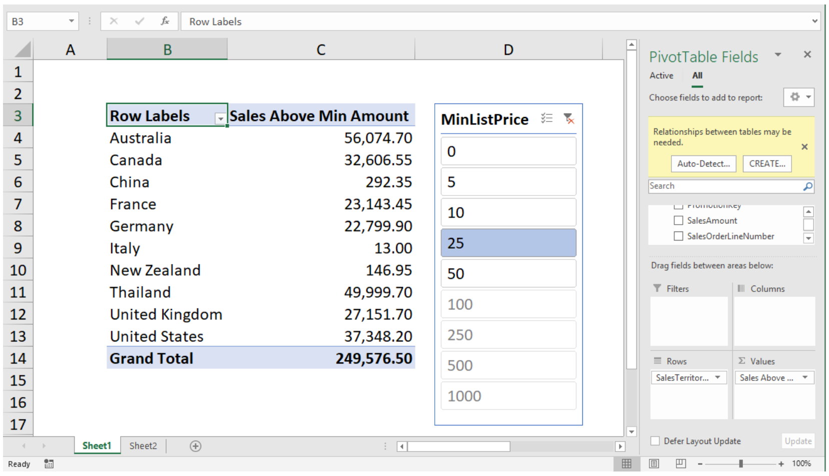
Insert a slicer for **MinListPrice** and add the **Sales Above Min Amount** into the 'Values' area of the PivotTable.



Row Labels	Sales Above Min Amount
Australia	402,474.75
Canada	231,539.60
China	1,423.85
France	167,266.10
Germany	169,588.10
Italy	763.75
New Zealand	720.70
Thailand	365,864.60
United Kingdom	207,991.25
United States	272,965.85
Grand Total	**1,820,598.55**

MinListPrice: 0 (selected), 5, 10, 25, 50, 100, 250, 500, 1000

Selecting different price points in the slicer will highlight the amount of **Sales** made where the unit price is above the minimum price point.



Row Labels	Sales Above Min Amount
Australia	56,074.70
Canada	32,606.55
China	292.35
France	23,143.45
Germany	22,799.90
Italy	13.00
New Zealand	146.95
Thailand	49,999.70
United Kingdom	27,151.70
United States	37,348.20
Grand Total	**249,576.50**

MinListPrice: 0, 5, 10, 25 (selected), 50, 100, 250, 500, 1000

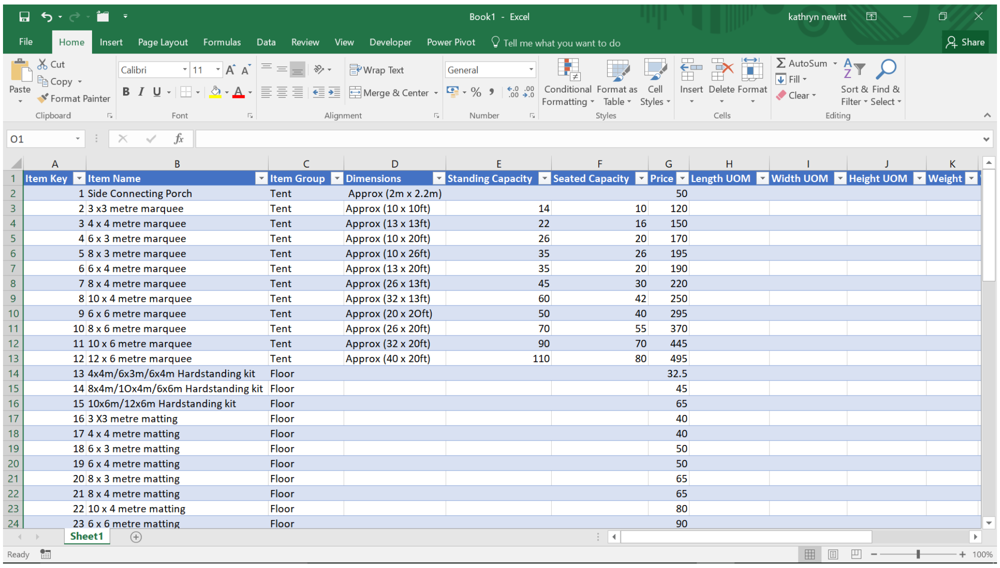
Interesting? We can apply the greater than or even the equals to operator to this filter.

More *Power Pivot Principles* next month.

Power Query Pointers

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

Clearly, renaming one column name is not difficult to do manually, but this month, we are looking at the situation where we have the column names below:



	A	B	C	D	E	F	G	H	I	J	K
1	Item Key	Item Name	Item Group	Dimensions	Standing Capacity	Seated Capacity	Price	Length UOM	Width UOM	Height UOM	Weight
2	1	Side Connecting Porch	Tent	Approx (2m x 2.2m)			50				
3	2	3 x3 metre marquee	Tent	Approx (10 x 10ft)	14	10	120				
4	3	4 x 4 metre marquee	Tent	Approx (13 x 13ft)	22	16	150				
5	4	6 x 3 metre marquee	Tent	Approx (10 x 20ft)	26	20	170				
6	5	8 x 3 metre marquee	Tent	Approx (10 x 26ft)	35	26	195				
7	6	6 x 4 metre marquee	Tent	Approx (13 x 20ft)	35	20	190				
8	7	8 x 4 metre marquee	Tent	Approx (26 x 13ft)	45	30	220				
9	8	10 x 4 metre marquee	Tent	Approx (32 x 13ft)	60	42	250				
10	9	6 x 6 metre marquee	Tent	Approx (20 x 2Oft)	50	40	295				
11	10	8 x 6 metre marquee	Tent	Approx (26 x 20ft)	70	55	370				
12	11	10 x 6 metre marquee	Tent	Approx (32 x 20ft)	90	70	445				
13	12	12 x 6 metre marquee	Tent	Approx (40 x 20ft)	110	80	495				
14	13	4x4m/6x3m/6x4m Hardstanding kit	Floor				32.5				
15	14	8x4m/1Ox4m/6x6m Hardstanding kit	Floor				45				
16	15	10x6m/12x6m Hardstanding kit	Floor				65				
17	16	3 X3 metre matting	Floor				40				
18	17	4 x 4 metre matting	Floor				40				
19	18	6 x 3 metre matting	Floor				50				
20	19	6 x 4 metre matting	Floor				50				
21	20	8 x 3 metre matting	Floor				65				
22	21	8 x 4 metre matting	Floor				65				
23	22	10 x 4 metre matting	Floor				80				
24	23	6 x 6 metre matting	Floor				90				

Many of the column headings have spaces in them, and we want to put an underscore (_) in instead, to prevent any problems that spaces could cause. Let's create a 'New Query' in the 'Get and Transform' section on the Data tab, and choose to create the query 'From Table'. There is a function in **M** devoted to transforming column names:

Table.TransformColumnNames(table as table, **nameGenerator** as function, optional **options** as nullable record) as table

The description for this function in Microsoft's Power Query help pages is:

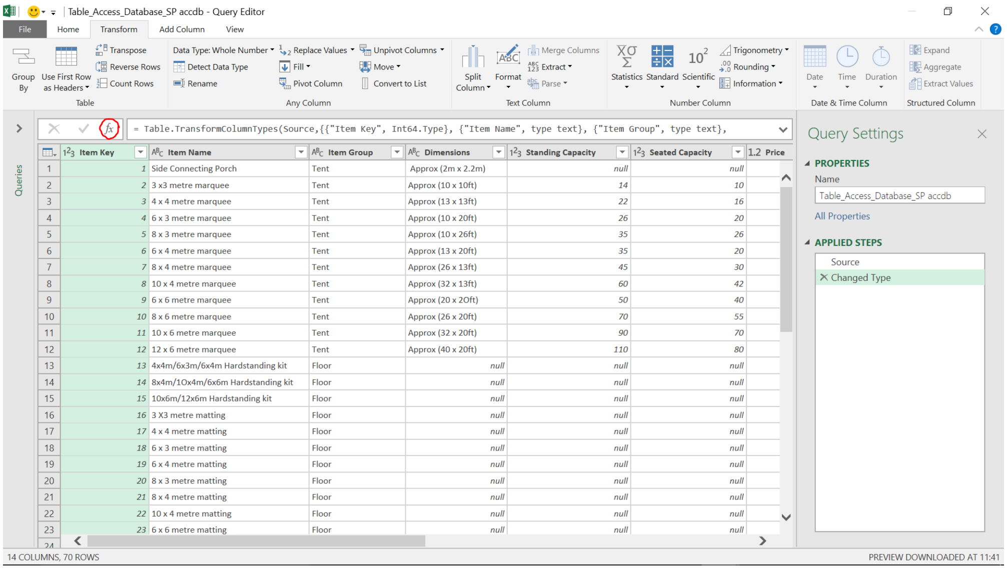
Transforms column names by using the given **nameGenerator** function. Valid options consist of **MaxLength**, which specifies the maximum length of new column names. If the given function results with a longer column name, the long name will be trimmed. The comparer is used to control the comparison while generating new column names. Comparers can be used to provide case insensitive or culture and locale aware comparisons. The following built in comparers are available in the formula language:

- **Comparer.Ordinal:** used to perform an exact ordinal comparison
- **Comparer.OrdinalIgnoreCase:** used to perform an exact ordinal case-insensitive **Comparison**
- **Comparer.FromCulture:** used to perform a culture aware comparison.

The best way to see how this works is with an example. To replace the spaces in my column headings with an underscore, I use the following **M** language:

Table.TransformColumnNames(Source, each Text.Replace(_, " ", "_"))

Essentially, this goes through my source data and each time it comes across a column name, it looks for a space and replaces it with an underscore. We need to apply this to our query.



```
= Table.TransformColumnTypes(Source,{{"Item Key", Int64.Type}, {"Item Name", type text}, {"Item Group", type text},
```

	Item Key	Item Name	Item Group	Dimensions	Standing Capacity	Seated Capacity	Price
1	1	Side Connecting Porch	Tent	Approx (2m x 2.2m)	null	null	
2	2	3 x3 metre marquee	Tent	Approx (10 x 10ft)	14	10	
3	3	4 x 4 metre marquee	Tent	Approx (13 x 13ft)	22	16	
4	4	6 x 3 metre marquee	Tent	Approx (10 x 20ft)	26	20	
5	5	8 x 3 metre marquee	Tent	Approx (10 x 26ft)	35	26	
6	6	6 x 4 metre marquee	Tent	Approx (13 x 20ft)	35	20	
7	7	8 x 4 metre marquee	Tent	Approx (26 x 13ft)	45	30	
8	8	10 x 4 metre marquee	Tent	Approx (32 x 13ft)	60	42	
9	9	6 x 6 metre marquee	Tent	Approx (20 x 2Oft)	50	40	
10	10	8 x 6 metre marquee	Tent	Approx (26 x 20ft)	70	55	
11	11	10 x 6 metre marquee	Tent	Approx (32 x 20ft)	90	70	
12	12	12 x 6 metre marquee	Tent	Approx (40 x 20ft)	110	80	
13	13	4x4m/6x3m/6x4m Hardstanding kit	Floor	null	null	null	
14	14	8x4m/1Ox4m/6x6m Hardstanding kit	Floor	null	null	null	
15	15	10x6m/12x6m Hardstanding kit	Floor	null	null	null	
16	16	3 X3 metre matting	Floor	null	null	null	
17	17	4 x 4 metre matting	Floor	null	null	null	
18	18	6 x 3 metre matting	Floor	null	null	null	
19	19	6 x 4 metre matting	Floor	null	null	null	
20	20	8 x 3 metre matting	Floor	null	null	null	
21	21	8 x 4 metre matting	Floor	null	null	null	
22	22	10 x 4 metre matting	Floor	null	null	null	
23	23	6 x 6 metre matting	Floor	null	null	null	

Query Settings

PROPERTIES

Name: Table_Access_Database_SP accdb

All Properties

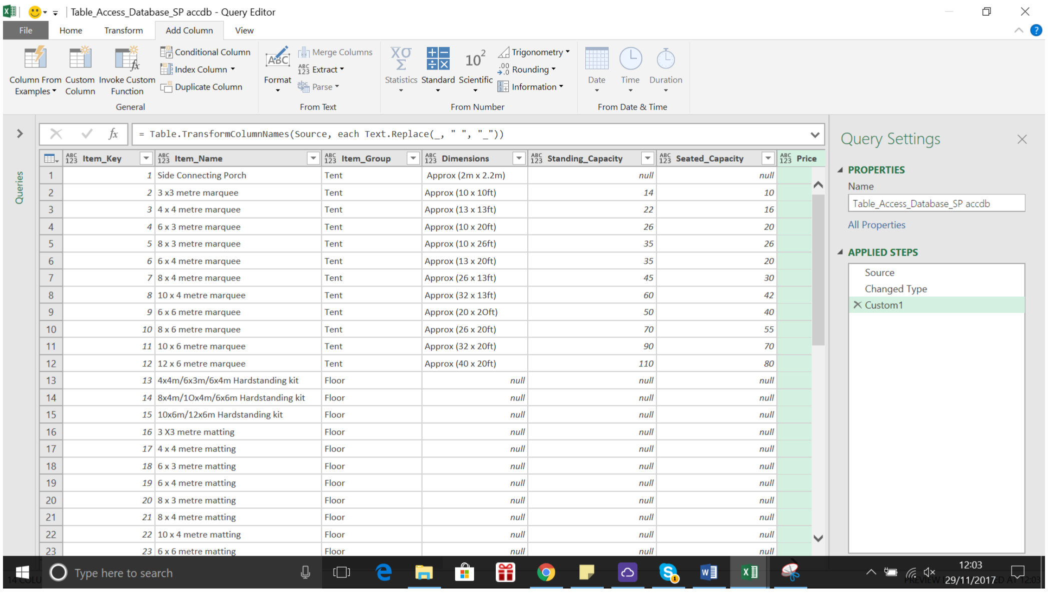
APPLIED STEPS

- Source
- Changed Type

14 COLUMNS, 70 ROWS

PREVIEW DOWNLOADED AT 11:41

If we use the **fx** button, as highlighted above, we can create a new step to transform my columns. Let’s enter this **M** function and use the tick icon to create a new step.



```
= Table.TransformColumnNames(Source, each Text.Replace(_, " ", "_"))
```

	Item_Key	Item_Name	Item_Group	Dimensions	Standing_Capacity	Seated_Capacity	Price
1	1	Side Connecting Porch	Tent	Approx (2m x 2.2m)	null	null	
2	2	3 x3 metre marquee	Tent	Approx (10 x 10ft)	14	10	
3	3	4 x 4 metre marquee	Tent	Approx (13 x 13ft)	22	16	
4	4	6 x 3 metre marquee	Tent	Approx (10 x 20ft)	26	20	
5	5	8 x 3 metre marquee	Tent	Approx (10 x 26ft)	35	26	
6	6	6 x 4 metre marquee	Tent	Approx (13 x 20ft)	35	20	
7	7	8 x 4 metre marquee	Tent	Approx (26 x 13ft)	45	30	
8	8	10 x 4 metre marquee	Tent	Approx (32 x 13ft)	60	42	
9	9	6 x 6 metre marquee	Tent	Approx (20 x 2Oft)	50	40	
10	10	8 x 6 metre marquee	Tent	Approx (26 x 20ft)	70	55	
11	11	10 x 6 metre marquee	Tent	Approx (32 x 20ft)	90	70	
12	12	12 x 6 metre marquee	Tent	Approx (40 x 20ft)	110	80	
13	13	4x4m/6x3m/6x4m Hardstanding kit	Floor	null	null	null	
14	14	8x4m/1Ox4m/6x6m Hardstanding kit	Floor	null	null	null	
15	15	10x6m/12x6m Hardstanding kit	Floor	null	null	null	
16	16	3 X3 metre matting	Floor	null	null	null	
17	17	4 x 4 metre matting	Floor	null	null	null	
18	18	6 x 3 metre matting	Floor	null	null	null	
19	19	6 x 4 metre matting	Floor	null	null	null	
20	20	8 x 3 metre matting	Floor	null	null	null	
21	21	8 x 4 metre matting	Floor	null	null	null	
22	22	10 x 4 metre matting	Floor	null	null	null	
23	23	6 x 6 metre matting	Floor	null	null	null	

Name: Table_Access_Database_SP accdb

Applied steps:
- Source
- Changed Type
- Custom1

All the columns have been updated with an underscore instead of a space and we can load the data into the spreadsheet with our newly named columns.

More next month!

Power BI Updates

The following update includes a new preview for the Power Automate visual – but that’s not all. The ongoing Previews (Small Multiples and DirectQuery for Power BI datasets and Azure Analysis Services) have been afforded some new features. There are also improvements to the shapes in Power BI Desktop and Microsoft has introduced a new (allegedly easier) way to share reports too.

The full list reads as follows:

Reporting

- Power Automate for Power BI (Preview)
- Small Multiples: padding controls and combo chart support (Preview)
- New and improved shapes
- Invert axis and continuous axis sorting

Modelling

- DirectQuery for Power BI datasets and Azure Analysis Services
- **CROSSFILTER** supports many-to-many relationships
- Performance improvements to **IF** and **SWITCH** functions

Data connectivity

- New Connector: Bloomberg Data and Analytics
- New Connector: SoftOne BI
- Text / CSV By Example now Generally Available
- Automatic Table Detection from Excel files now Generally Available
- Automatic Table Detection from JSON files now Generally Available

Service

- Easy report sharing via links
- Improving the ‘Create on top of published dataset’ experience
- New capabilities in Deployment Pipelines
- Enhancements to Sensitivity Labels:
  - Sensitivity labels in Power BI embedded now Generally Available
  - Securing the full data journey from Azure to Office
  - Admin API to Set and Remove labels in Power BI

Visualisations

- New visuals
- Microsoft Charticulator visual
- Financial Reporting Matrix by Profitbase
- Merged Bar Chart by Nova Silva

Developers

- API v3.6.0 is now available

Other

- Simplifying collection of diagnostic information.

Let’s go through each in turn.

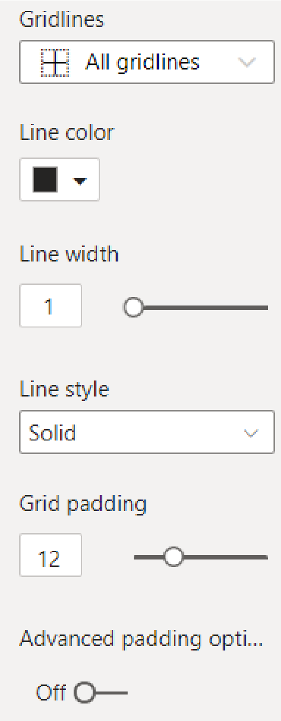
Power Automate for Power BI (Preview)

This updates sees a new Power Automate visual. Microsoft has reacted to end users stating that they want to act on the insights they find when exploring a Power BI report. Many times, the action happens outside of the product (*e.g.* send a note, create a task, copy the data). Additionally, developers want built-in automation for their users but need to ensure that it’s within the context of the end-user’s selection. With this new Power Automate visual, end-users may now run an automated flow all within a Power BI report. Furthermore, the executed flow can be data contextual, meaning that the Flow inputs can be dynamic based upon the filters set by the end-user.

Small Multiples: padding controls and combo chart support (Preview)

This update sees the addition of two new features for the Small Multiples Preview: padding controls and support for the line and clustered / stacked column charts (combo charts).

You'll find padding controls in the 'Grid layout' card in the Formatting pane, underneath the Gridlines controls:



Gridlines

All gridlines

Line color

Line width

1

Line style

Solid

Grid padding

12

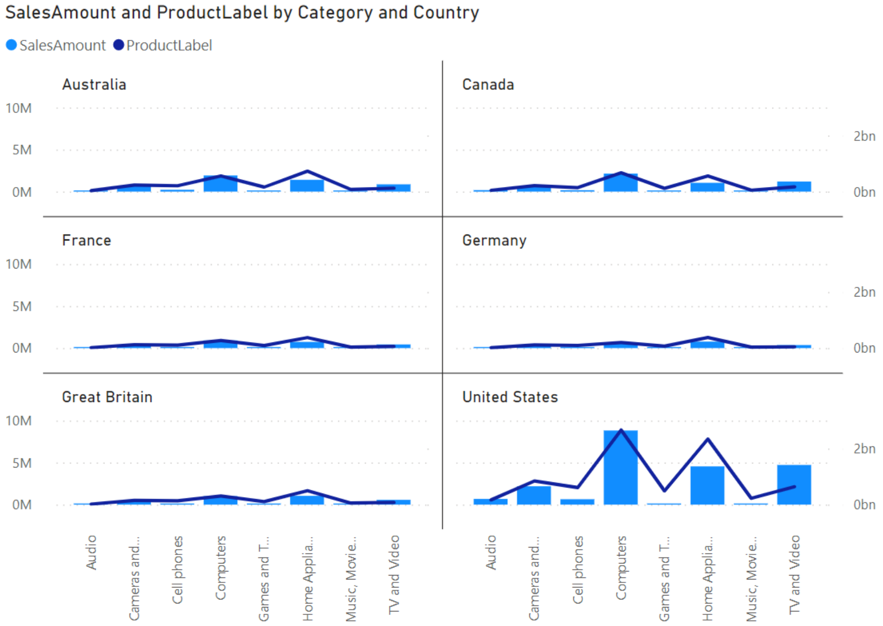
Advanced padding opti...

Off

Adjusting the 'Grid padding' slider will change the padding around your Small Multiples. The effects of this slider will vary automatically based on which pattern of gridlines you have selected. For example, the slider will not affect the padding along the outer edge of the small multiples grid, unless you have the grid's outside borders switched on.

For more granular control over the padding in the visual, turn on 'Advanced padding options' to replace the single grid padding slider with four new sliders: inner column padding, outer column padding, inner row padding and outer row padding. Adjusting these sliders will always affect the padding between multiples in the same way, regardless of your choice in gridlines.

Additionally, combo charts now also support Small Multiples:



One potential difference between this and non-Small Multiples combo charts is that here, Power BI will never automatically merge the column and line chart axes. However, you can still merge them yourself using the 'Show secondary' option in the Formatting pane.

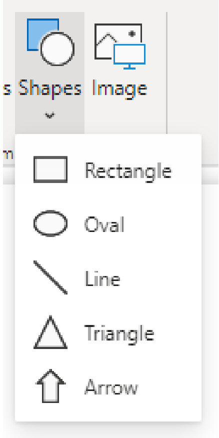
It should also be noted that the 'Show as a table' context menu option for combo charts may not work in every situation, and that dynamic format strings currently won't work in line chart data labels or ToolTips. However, fixes are underway for the next update.

New and improved shapes

Shapes in Power BI now include many more shapes, more formatting options, new default styling and the ability to upgrade any old shapes in your reports.

Previously in the old shape experience, you could choose the following shapes from the Ribbon:

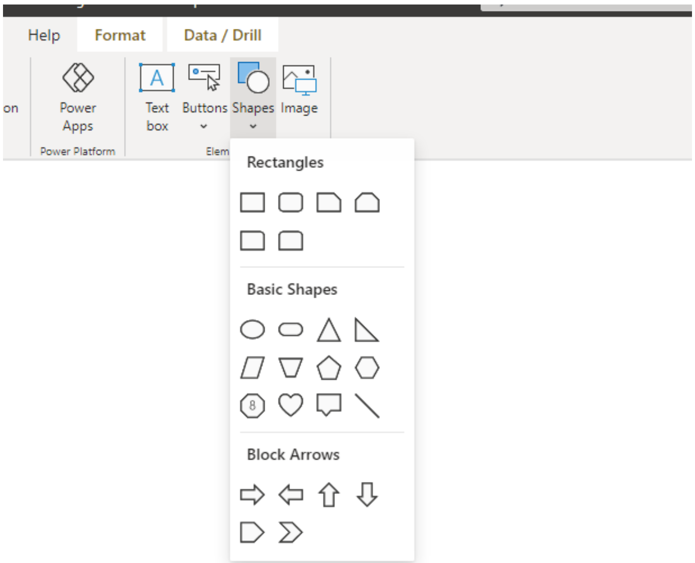
- Rectangle
- Oval
- Line
- Isosceles Triangle
- Arrow (up).



Now there are more:

- Rectangle
- Rounded Rectangle
- Tab: Single Corner Snipped
- Tab: Top Corners Snipped
- Tab: Single Corner Rounded
- Tab: Top Corners Rounded
- Oval
- Pill
- Isosceles Triangle
- Right Triangle
- Parallelogram
- Trapezoid
- Pentagon
- Hexagon
- Octagon
- Heart
- Speech bubble
- Line
- Arrow (left, right, up, down)
- Arrow: Pentagon
- Arrow: Chevron.

Any of these new shapes may be added from the Ribbon by navigating to **Insert -> Shapes:**



In addition to new shapes, this release also includes new formatting options such as:

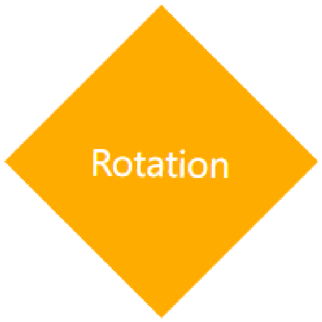
- Text options

- Round edges for rounded corners

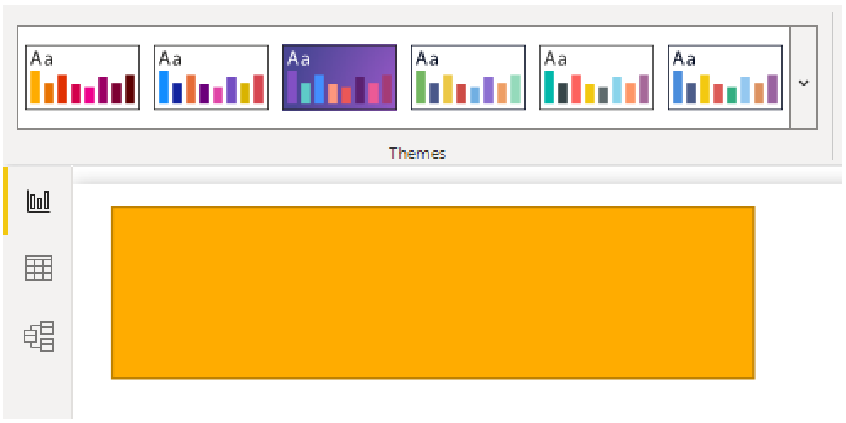
- Shape drop Shadow effects

- Shape Glow effects

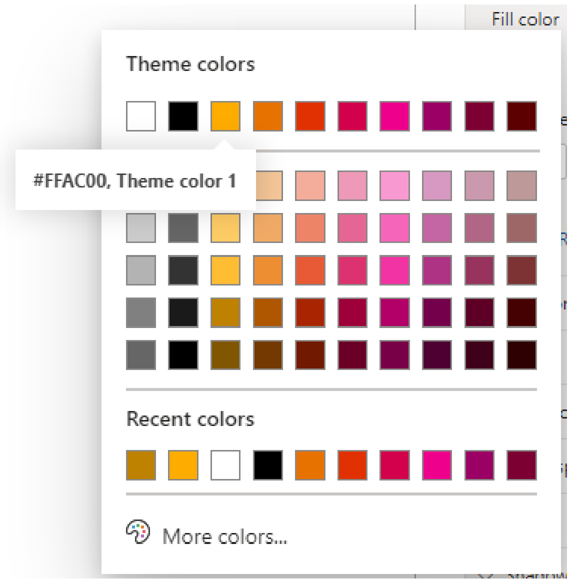
- Shape rotation and Text rotation.



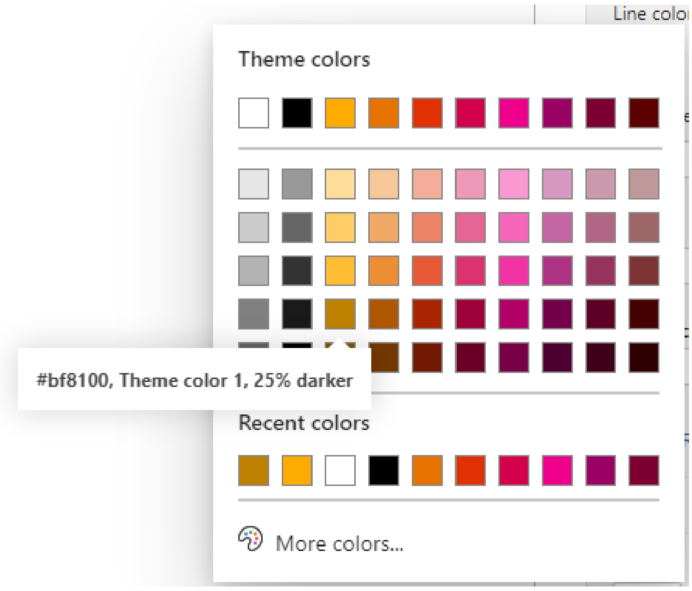
You may also notice that these shapes have a new default style for the fill colour, outline colour and outline weight.



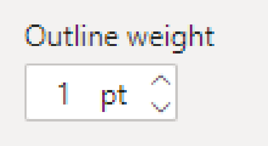
The Fill colour of new shapes will default to ‘Theme color 1’ *(sic)*:



The Outline colour of the new shape will default to ‘Theme color 1, 25% darker’:

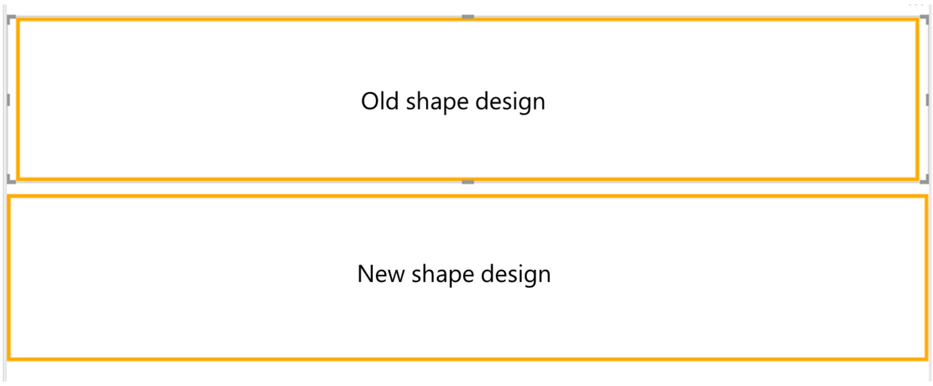


Additionally, the Outline weight will default to ‘1 pt’:


Outline weight

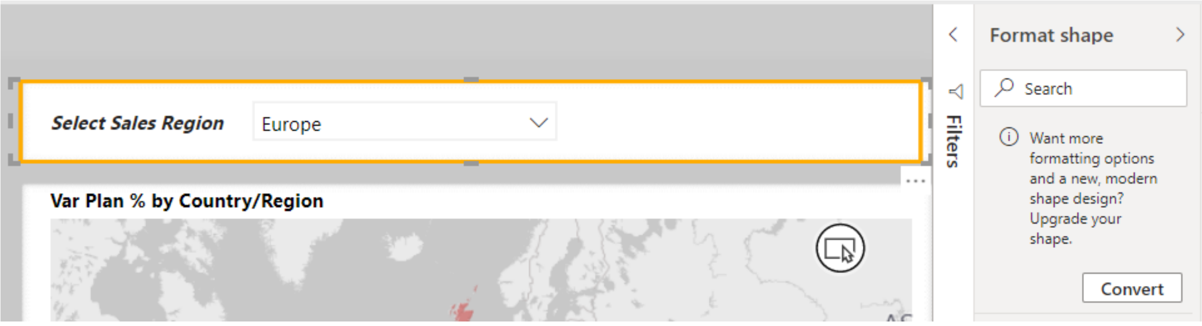
1 pt

Also, the size of the shape has been maximised relative to its container for the new design. Here’s an example of the rectangle shape, and you will notice the old rectangle design has additional padding on the left and right side of the border:



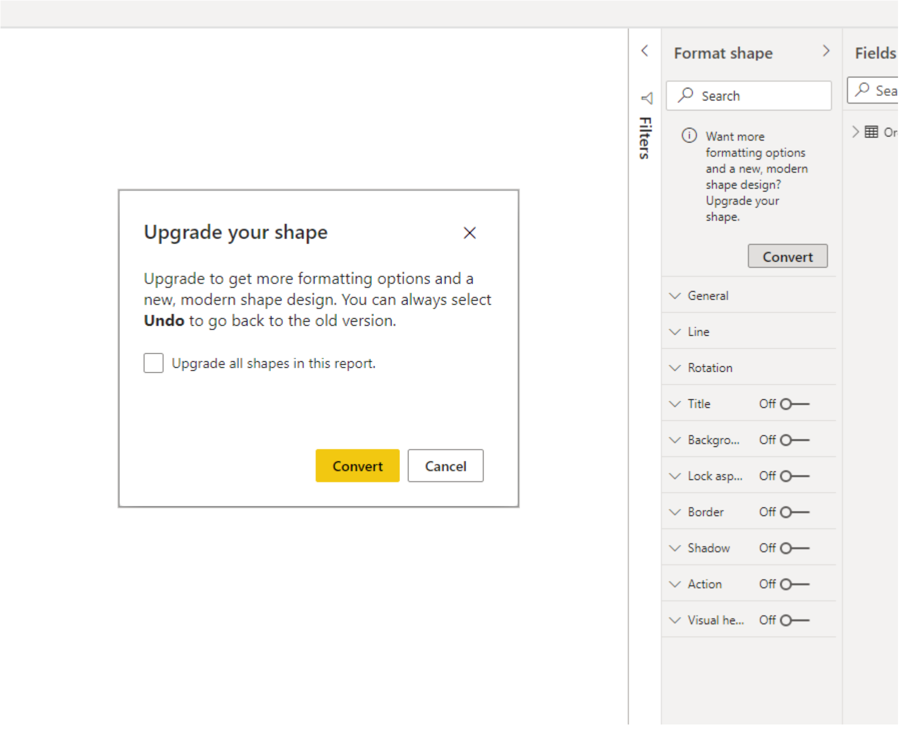
Note that after converting your rectangle shapes, you may want to make slight tweaks to the size of shape because the new shape design doesn't have the additional padding that the old rectangle had.

Also, if you have existing shapes in your report, you can convert them to the new shape design by selecting the shape and clicking Convert in the 'Format shape' pane. You can also use **CTRL + Click** to select multiple shapes at once and convert them simultaneously.



Want more formatting options and a new, modern shape design? Upgrade your shape.

Once you click Convert, you'll also have the option to 'Upgrade all shapes in this report', not just the selected shape(s):


Upgrade your shape

Upgrade to get more formatting options and a new, modern shape design. You can always select **Undo** to go back to the old version.

□ Upgrade all shapes in this report.

Invert axis and continuous axis sorting

This update introduces a new capability in line, bar, column, area and combo charts to invert the value axis. This allows you to reverse the direction in which the axis is rendered, with positive going down and negative going up. You can find the invert axis toggle in the associated 'Value axis' card in the Formatting pane.

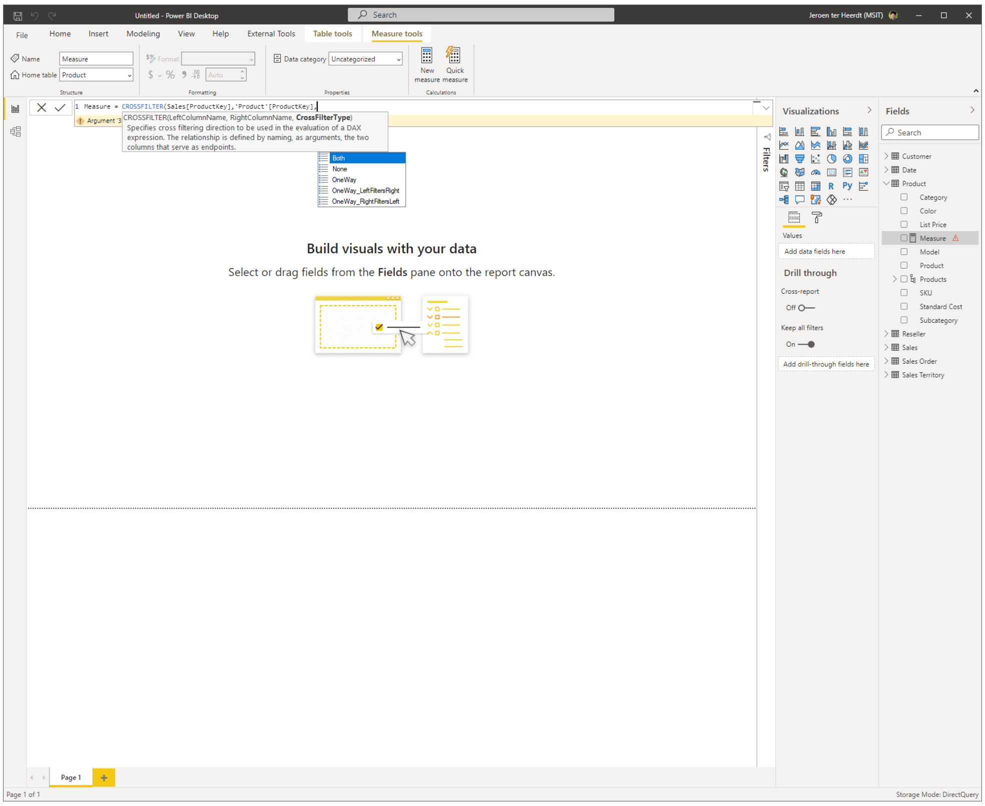
For the record, apparently the next release will see the release of a similar functionality to reverse continuous fields in the primary axis. You will be able to sort continuous axes by ascending or descending, enabling use cases like showing the most recent data first.

DirectQuery for Power BI datasets and Azure Analysis Services

This release adds the capability to connect to a perspective when making a DirectQuery connection an Azure Analysis Services model, assuming there is one available.

CROSSFILTER supports many-to-many relationships

Microsoft has added support for many-to-many relationships to the **CROSSFILTER** function. The **CrossFilterType** parameter allows you to specify how you want the filter to flow. Previously, you could choose Both, None or OneWay. For a many-to-many relationship, however, OneWay is not enough control as the filter could go either way. Therefore, this update has added OneWay_LeftFiltersRight and OneWay_RightFiltersLeft to provide more control over the direction the filter flows in this scenario.



```
1 Measure = CROSSFILTER(Sales[ProductKey],'Product'[ProductKey],
```

CROSSFILTER(LeftColumnName, RightColumnName, **CrossFilterType**)
Specifies cross filtering direction to be used in the evaluation of a DAX expression. The relationship is defined by naming, as arguments, the two columns that serve as endpoints.

Performance improvements to IF and SWITCH functions

A performance improvement has been made to **SWITCH** function with many branches and deeply nested **IF** functions, to eliminate branches which are not selected by user filters or slicers earlier in the calculation pipeline. The DAX pattern for the branch conditions covered by the optimisation is matching **SELECTEDVALUE(column), VALUES(column), MIN(column)** or **MAX(column)** to values of the column.

New Connector: Bloomberg Data and Analytics

This update sees the release of the Bloomberg Data and Analytics Power Query connector in Power BI Desktop.

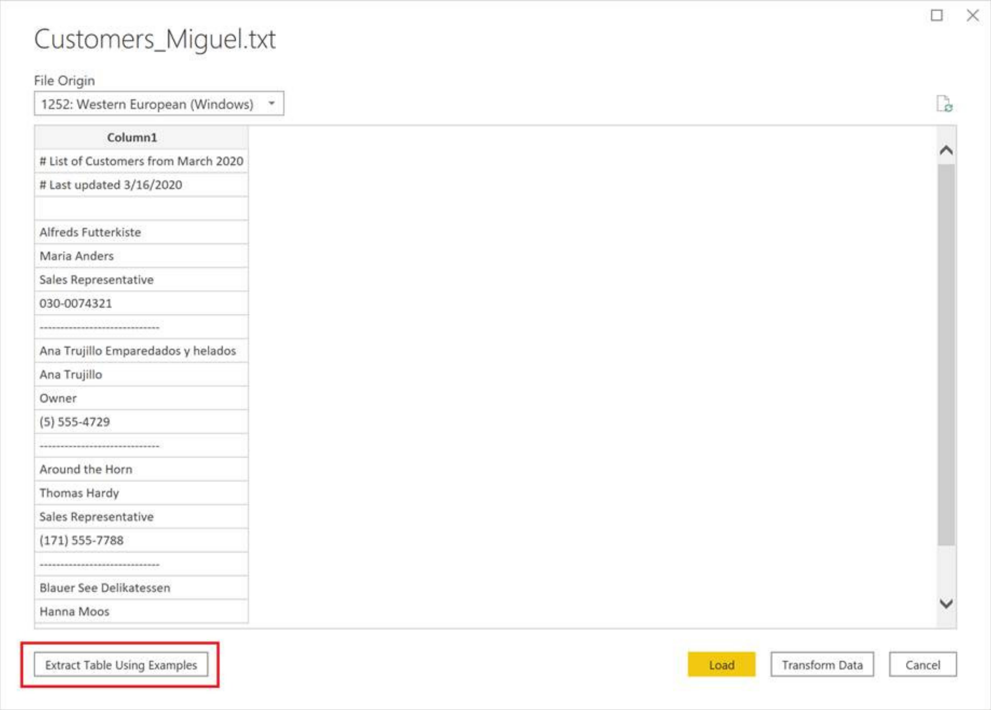
New Connector: Bloomberg Data and Analytics

This update also sees the release of the SoftOne BI Power Query connector in Power BI Desktop.

Text / CSV By Example now Generally Available

'Text / CSV By Example' in Power Query is now Generally Available. This feature makes it extremely easy for users to extract data from Text or CSV files without having to think about the data transformations to apply, rather just providing a set of sample values to extract and let Power Query infer the transformations to achieve it.

When using the Text / CSV connector, users will see a new option to 'Extract Table Using Examples' on the bottom-left corner of the file preview dialog:



Customers_Miguel.txt
File Origin
1252: Western European (Windows)
Extract Table Using Examples
Load
Transform Data
Cancel

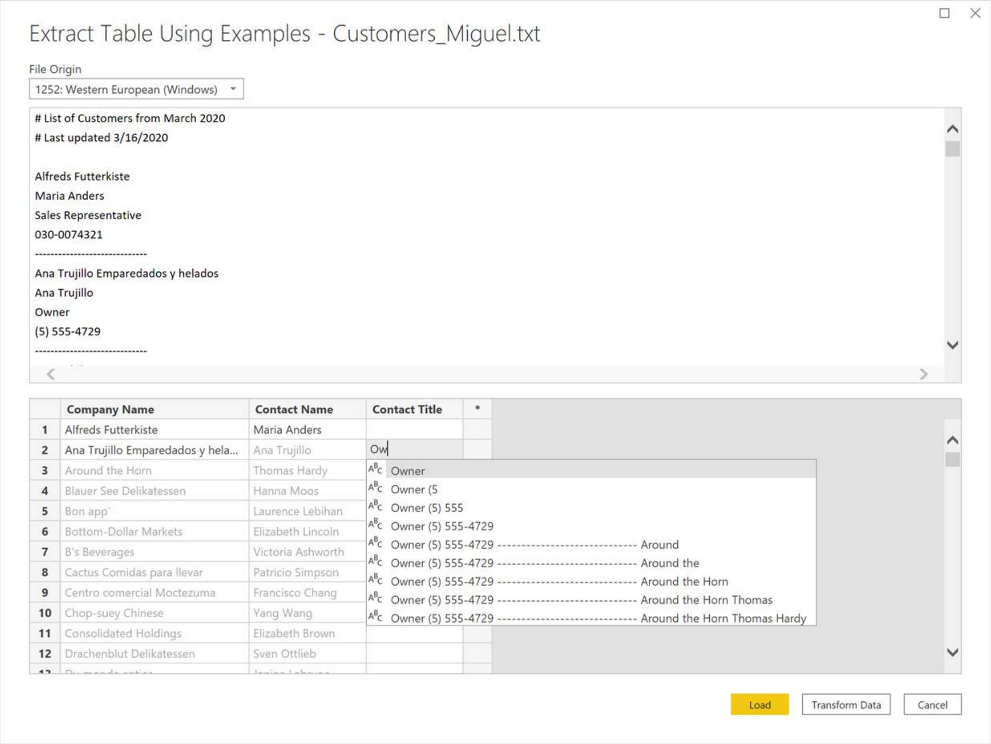
Column1
List of Customers from March 2020
Last updated 3/16/2020
Alfreds Futterkiste
Maria Anders
Sales Representative
030-0074321

Ana Trujillo Emparedados y helados
Ana Trujillo
Owner
(5) 555-4729

Around the Horn
Thomas Hardy
Sales Representative
(171) 555-7788

Blauer See Delikatessen
Hanna Moos

Upon clicking that new button, users will be taken into the ‘Extract Table Using Examples’ experience allowing them to specify sample output values for the data they would like to extract from their Text / CSV file.



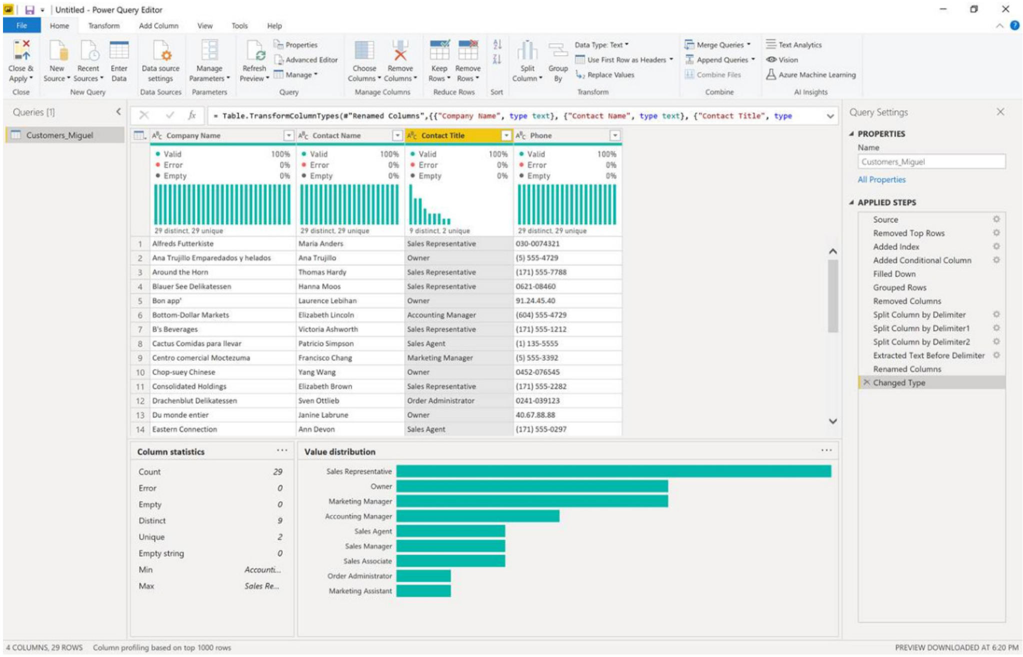
Extract Table Using Examples - Customers_Miguel.txt
File Origin
1252: Western European (Windows)
Load
Transform Data
Cancel

```
# List of Customers from March 2020
# Last updated 3/16/2020

Alfreds Futterkiste
Maria Anders
Sales Representative
030-0074321
----------------------------
Ana Trujillo Emparedados y helados
Ana Trujillo
Owner
(5) 555-4729
----------------------------
```

	Company Name	Contact Name	Contact Title	*
1	Alfreds Futterkiste	Maria Anders		
2	Ana Trujillo Emparedados y hela...	Ana Trujillo	Ow	
3	Around the Horn	Thomas Hardy		
4	Blauer See Delikatessen	Hanna Moos		
5	Bon app'	Laurence Lebihan		
6	Bottom-Dollar Markets	Elizabeth Lincoln		
7	B's Beverages	Victoria Ashworth		
8	Cactus Comidas para llevar	Patricio Simpson		
9	Centro comercial Moctezuma	Francisco Chang		
10	Chop-suey Chinese	Yang Wang		
11	Consolidated Holdings	Elizabeth Brown		
12	Drachenblut Delikatessen	Sven Ottlieb		

```
Owner
Owner (5
Owner (5) 555
Owner (5) 555-4729
Owner (5) 555-4729 ---------------------------- Around
Owner (5) 555-4729 ---------------------------- Around the
Owner (5) 555-4729 ---------------------------- Around the Horn
Owner (5) 555-4729 ---------------------------- Around the Horn Thomas
Owner (5) 555-4729 ---------------------------- Around the Horn Thomas Hardy
```

Once users have completed table construction, they can click Load / Transform to complete the ‘Get Data flow’. Notice how the resulting queries contain a detailed breakdown of all the steps that were inferred for the data extraction, which are just regular query steps that can be customised as needed.



```
= Table.TransformColumnTypes(#"Renamed Columns",{{"Company Name", type text}, {"Contact Name", type text}, {"Contact Title", type
```

	Company Name	Contact Name	Contact Title	Phone
	Valid 100% Error 0% Empty 0% 29 distinct, 29 unique	Valid 100% Error 0% Empty 0% 29 distinct, 29 unique	Valid 100% Error 0% Empty 0% 9 distinct, 2 unique	Valid 100% Error 0% Empty 0% 29 distinct, 29 unique
1	Alfreds Futterkiste	Maria Anders	Sales Representative	030-0074321
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Owner	(5) 555-4729
3	Around the Horn	Thomas Hardy	Sales Representative	(171) 555-7788
4	Blauer See Delikatessen	Hanna Moos	Sales Representative	0621-08460
5	Bon app'	Laurence Lebihan	Owner	91.24.45.40
6	Bottom-Dollar Markets	Elizabeth Lincoln	Accounting Manager	(604) 555-4729
7	B's Beverages	Victoria Ashworth	Sales Representative	(171) 555-1212
8	Cactus Comidas para llevar	Patricio Simpson	Sales Agent	(1) 135-5555
9	Centro comercial Moctezuma	Francisco Chang	Marketing Manager	(5) 555-3392
10	Chop-suey Chinese	Yang Wang	Owner	0452-076545
11	Consolidated Holdings	Elizabeth Brown	Sales Representative	(171) 555-2282
12	Drachenblut Delikatessen	Sven Ottlieb	Order Administrator	0241-039123
13	Du monde entier	Janine Labrune	Owner	40.67.88.88
14	Eastern Connection	Ann Devon	Sales Agent	(171) 555-0297

Column statistics

Count	29
Error	0
Empty	0
Distinct	9
Unique	2
Empty string	0
Min	Accounti...
Max	Sales Re...

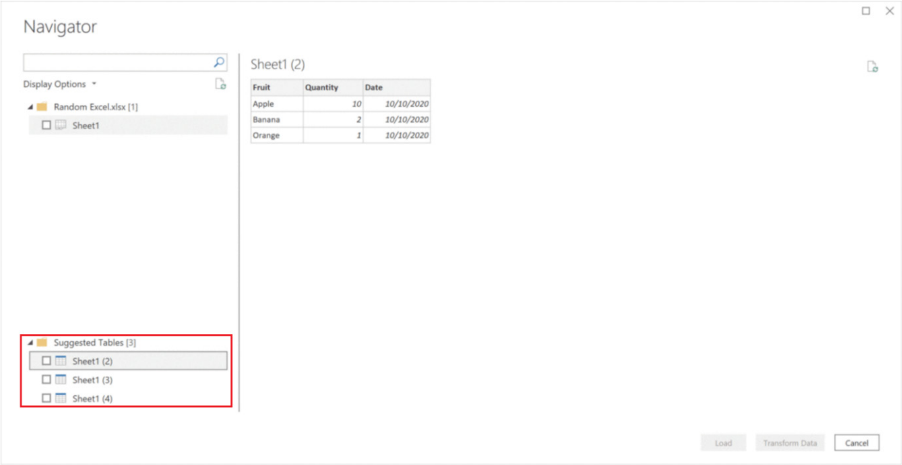
APPLIED STEPS
- Source
- Removed Top Rows
- Added Index
- Added Conditional Column
- Filled Down
- Grouped Rows
- Removed Columns
- Split Column by Delimiter
- Split Column by Delimiter1
- Split Column by Delimiter2
- Extracted Text Before Delimiter
- Renamed Columns
- Changed Type

4 COLUMNS, 29 ROWS Column profiling based on top 1000 rows PREVIEW DOWNLOADED AT 6:20 PM

Automatic Table Detection from Excel files now Generally Available

Excel is one of the most commonly used sources of data for reporting and analytics. A typical challenge with extracting data from Excel files is that the data very often is not formatted as tables so the task to "scrape" relevant data from an Excel spreadsheet can be extremely challenging and time consuming for users.

When using the Excel connector, this feature will automatically identify sections of each Excel spreadsheet to extract into a table and show them under a 'Suggested Tables' group in the Navigator.

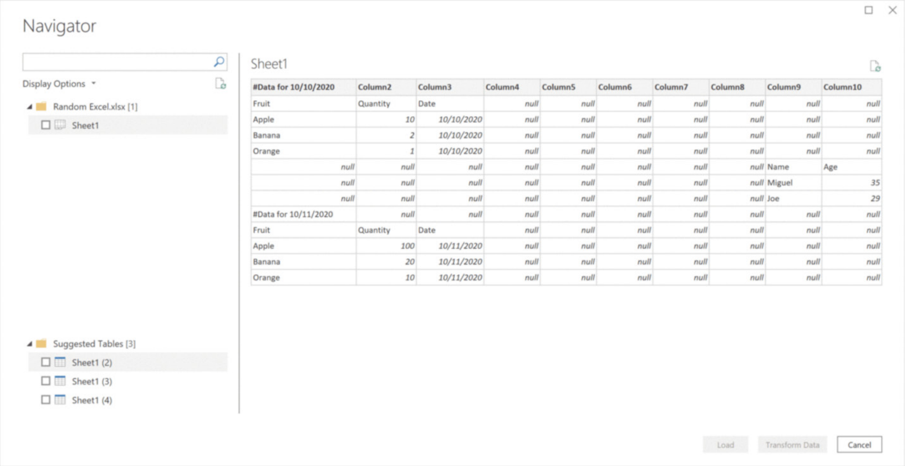


- Random Excel.xlsx [1]
  - □ Sheet1
- Suggested Tables [3]
  - □ Sheet1 (2)
  - □ Sheet1 (3)
  - □ Sheet1 (4)

Sheet1 (2)

Fruit	Quantity	Date
Apple	10	10/10/2020
Banana	2	10/10/2020
Orange	1	10/10/2020

Previously, unless data was formatted as Tables or Named Ranges in Excel, users had to scrape the relevant rows / columns with specific transforms (skip rows, remove columns, *etc.*) from worksheet objects (*e.g.* **Sheet1** in the example below).



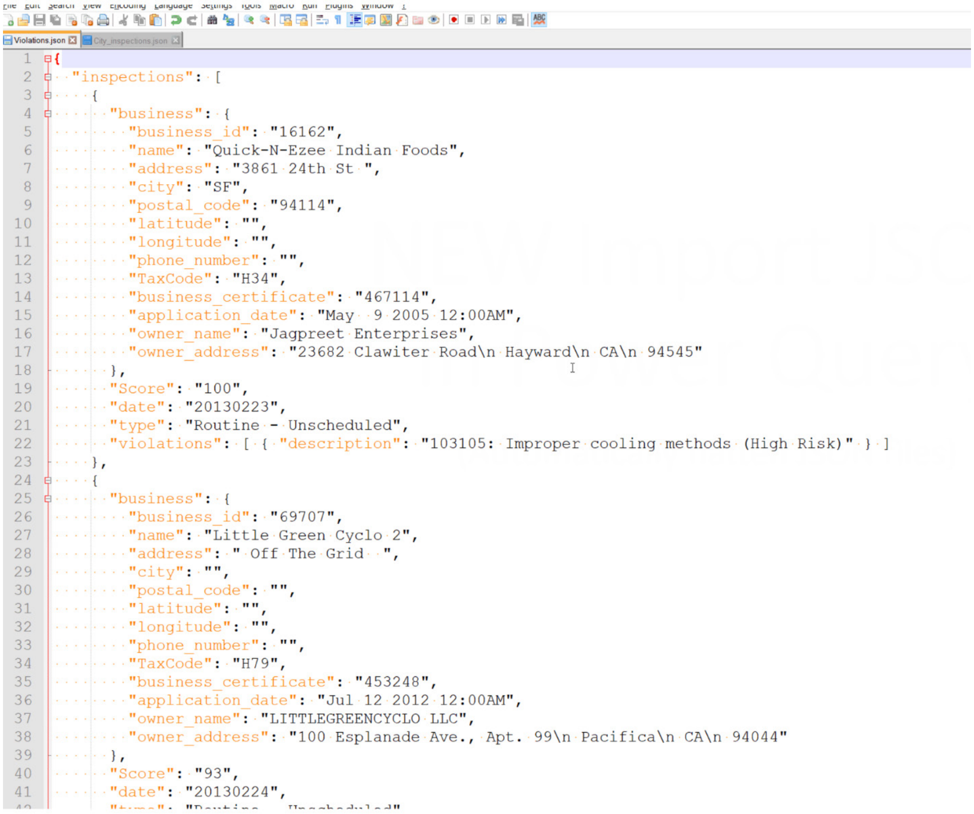
- Random Excel.xlsx [1]
  - □ Sheet1
- Suggested Tables [3]
  - □ Sheet1 (2)
  - □ Sheet1 (3)
  - □ Sheet1 (4)

Sheet1

#Data for 10/10/2020	Column2	Column3	Column4	Column5	Column6	Column7	Column8	Column9	Column10
Fruit	Quantity	Date	null	null	null	null	null	null	null
Apple	10	10/10/2020	null	null	null	null	null	null	null
Banana	2	10/10/2020	null	null	null	null	null	null	null
Orange	1	10/10/2020	null	null	null	null	null	null	null
null	null	null	null	null	null	null	null	Name	Age
null	null	null	null	null	null	null	null	Miguel	35
null	null	null	null	null	null	null	null	Joe	29
#Data for 10/11/2020	null	null	null	null	null	null	null	null	null
Fruit	Quantity	Date	null	null	null	null	null	null	null
Apple	100	10/11/2020	null	null	null	null	null	null	null
Banana	20	10/11/2020	null	null	null	null	null	null	null
Orange	10	10/11/2020	null	null	null	null	null	null	null

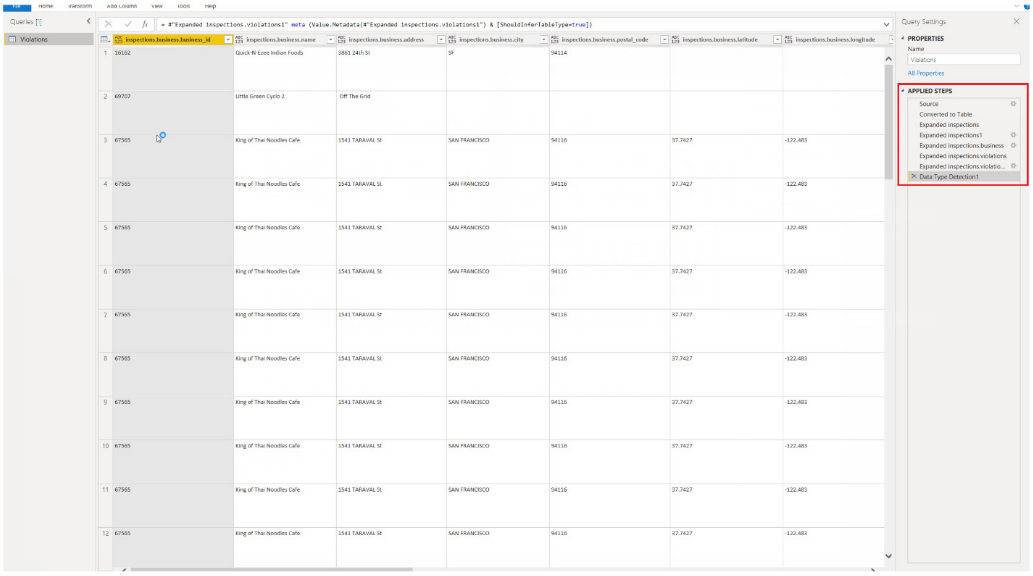
Automatic Table Detection from JSON files now Generally Available

Similar to the previous feature for Excel automatic table detection, importing data from JSON files (or Web APIs) can be equally challenging for end users. Here is an example of JSON file with multiple levels of nested data:


File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Violations.json
City_inspections.json

```
{
  "inspections": [
    {
      "business": {
        "business_id": "16162",
        "name": "Quick-N-Ezee Indian Foods",
        "address": "3861 24th St ",
        "city": "SF",
        "postal_code": "94114",
        "latitude": "",
        "longitude": "",
        "phone_number": "",
        "TaxCode": "H34",
        "business_certificate": "467114",
        "application_date": "May  9 2005 12:00AM",
        "owner_name": "Jagpreet Enterprises",
        "owner_address": "23682 Clawiter Road\n Hayward\n CA\n 94545"
      },
      "Score": "100",
      "date": "20130223",
      "type": "Routine - Unscheduled",
      "violations": [ { "description": "103105: Improper cooling methods (High Risk)" } ]
    },
    {
      "business": {
        "business_id": "69707",
        "name": "Little Green Cyclo 2",
        "address": " Off The Grid  ",
        "city": "",
        "postal_code": "",
        "latitude": "",
        "longitude": "",
        "phone_number": "",
        "TaxCode": "H79",
        "business_certificate": "453248",
        "application_date": "Jul 12 2012 12:00AM",
        "owner_name": "LITTLEGREENCYCLO LLC",
        "owner_address": "100 Esplanade Ave., Apt. 99\n Pacifica\n CA\n 94044"
      },
      "Score": "93",
      "date": "20130224",
      "type": "Routine - Unscheduled"
```

With the addition of Automatic Table Detection capabilities, when using the JSON connector, Power Query will automatically apply transformation steps to flatten the JSON data into a table. Previously, users had to flatten records / lists manually. This new feature also adds support for JSON lines (or newline-delimited JSON, where each line in the file is a JSON string).



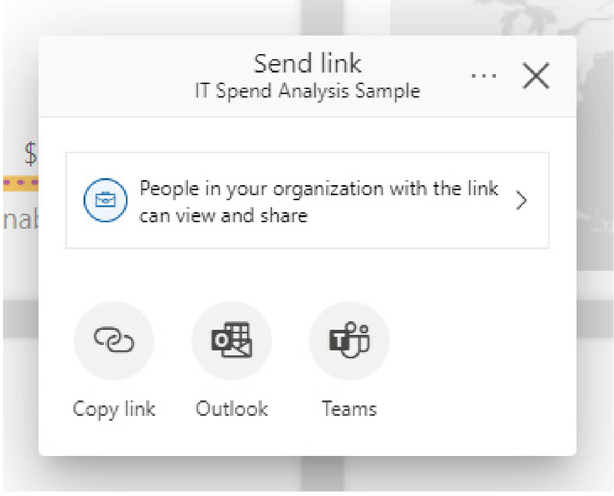
```
= #"Expanded inspections.violations1" meta (Value.Metadata(#"Expanded inspections.violations1") & [ShouldInferTableType=true])
```

	inspections.business.business_id	inspections.business.name	inspections.business.address	inspections.business.city	inspections.business.postal_code	inspections.business.latitude	inspections.business.longitude
1	16162	Quick-N-Ezee Indian Foods	3861 24th St	SF	94114		
2	69707	Little Green Cyclo 2	Off The Grid				
3	67565	King of Thai Noodles Cafe	1541 TARAVAL St	SAN FRANCISCO	94116	37.7427	-122.483
4	67565	King of Thai Noodles Cafe	1541 TARAVAL St	SAN FRANCISCO	94116	37.7427	-122.483
5	67565	King of Thai Noodles Cafe	1541 TARAVAL St	SAN FRANCISCO	94116	37.7427	-122.483
6	67565	King of Thai Noodles Cafe	1541 TARAVAL St	SAN FRANCISCO	94116	37.7427	-122.483
7	67565	King of Thai Noodles Cafe	1541 TARAVAL St	SAN FRANCISCO	94116	37.7427	-122.483
8	67565	King of Thai Noodles Cafe	1541 TARAVAL St	SAN FRANCISCO	94116	37.7427	-122.483
9	67565	King of Thai Noodles Cafe	1541 TARAVAL St	SAN FRANCISCO	94116	37.7427	-122.483
10	67565	King of Thai Noodles Cafe	1541 TARAVAL St	SAN FRANCISCO	94116	37.7427	-122.483
11	67565	King of Thai Noodles Cafe	1541 TARAVAL St	SAN FRANCISCO	94116	37.7427	-122.483
12	67565	King of Thai Noodles Cafe	1541 TARAVAL St	SAN FRANCISCO	94116	37.7427	-122.483

APPLIED STEPS
- Source
- Converted to Table
- Expanded inspections
- Expanded inspections1
- Expanded inspections.business
- Expanded inspections.violations
- Expanded inspections.violatio...
- Data Type Detection1

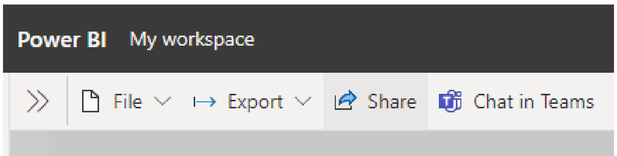
Easy report sharing via links

Microsoft has released a new sharing experience to support sharing reports via links. Previously, in the old experience, Power BI supported sharing reports by granting direct access. For the new experience, reports can be shared via links in addition to direct access sharing.

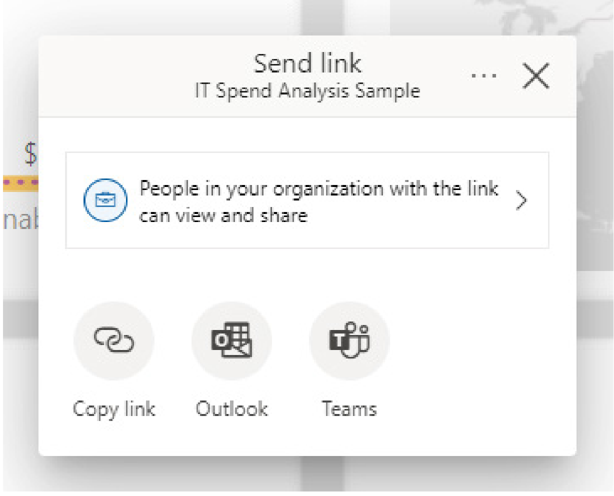

Send link
IT Spend Analysis Sample
People in your organization with the link can view and share
Copy link
Outlook
Teams

Currently, this new link sharing capability is available to reports only. However, Microsoft has also upgraded the share and grant access dialog for dashboards, paginated reports and datasets to match this new look.

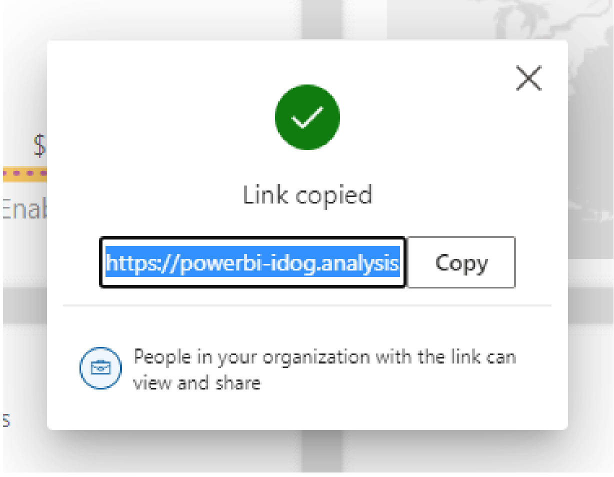
To copy or share links to a report, you can access this feature by navigating to Share:



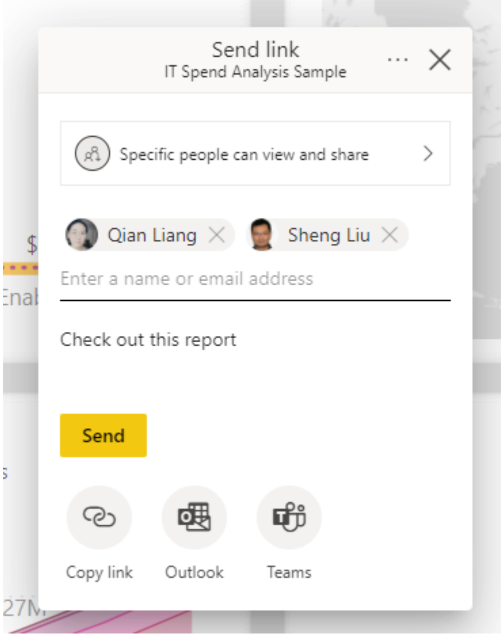
You'll see this new 'Send link' dialog that will allow you to copy the sharing link or share it via Outlook and Teams to people in your organisation:


Send link
IT Spend Analysis Sample
People in your organization with the link can view and share
Copy link
Outlook
Teams

Selecting 'Copy link' will automatically generate and copy a shareable link to your clipboard:


Link copied
https://powerbi-idog.analysis
Copy
People in your organization with the link can view and share

You can also choose to directly send the link to specific people or groups. Simply enter their name or email address, optionally type a message, and click Send.



Send link
IT Spend Analysis Sample

Specific people can view and share

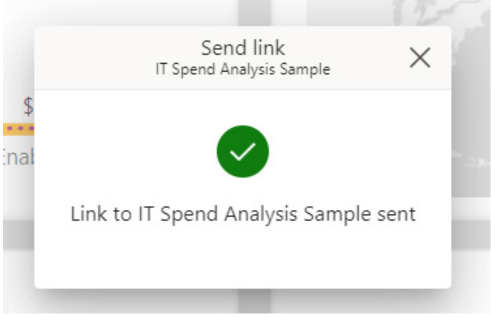
Qian Liang ✕ Sheng Liu ✕

Enter a name or email address

Check out this report

Send

Once you've click Send, the link will be sent via email to your recipients.

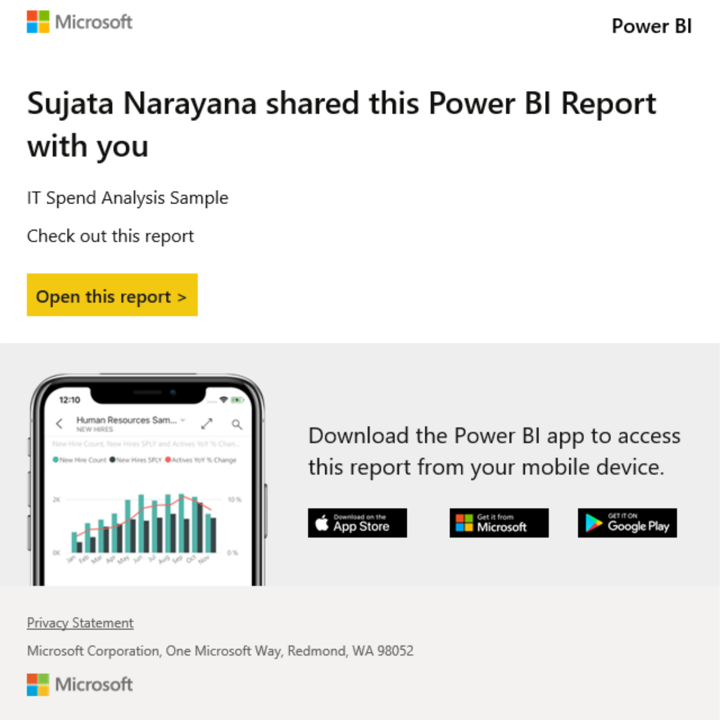


Send link
IT Spend Analysis Sample

✓

Link to IT Spend Analysis Sample sent

Once your recipients receive the email, they can click 'Open this report' and automatically get access to the report through the shareable link.



Microsoft

Power BI

Sujata Narayana shared this Power BI Report with you

IT Spend Analysis Sample

Check out this report

Open this report >

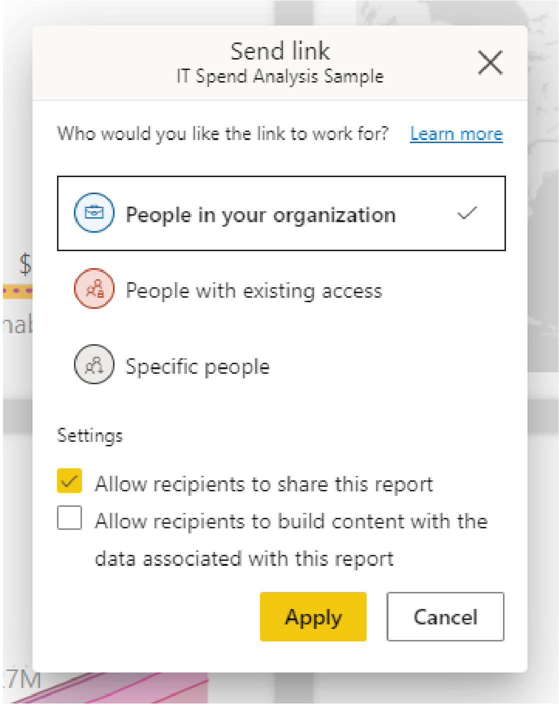
Download the Power BI app to access this report from your mobile device.

Privacy Statement

Microsoft Corporation, One Microsoft Way, Redmond, WA 98052

Microsoft

For ‘Link settings’, you may choose who may access your shared link, and what they can do with the report and associated data:



Send link
IT Spend Analysis Sample

Who would you like the link to work for? Learn more

People in your organization ✓

People with existing access

Specific people

Settings

☑ Allow recipients to share this report

□ Allow recipients to build content with the data associated with this report

Apply Cancel

For 'People in your organization' *(sic)*, this type of link can allow colleagues to access the report. This link will <u>not</u> work for external users (including guest users). You should only use this link when you want to share with someone in your organisation and are comfortable with them passing the link around to other people inside your organisation, but when you want to ensure that the link won’t work for external users.

For ‘People with existing access’, this type of link generates a URL to the report, but it does not give any access to the report. Use this if you just want to send a link to somebody who already has access.

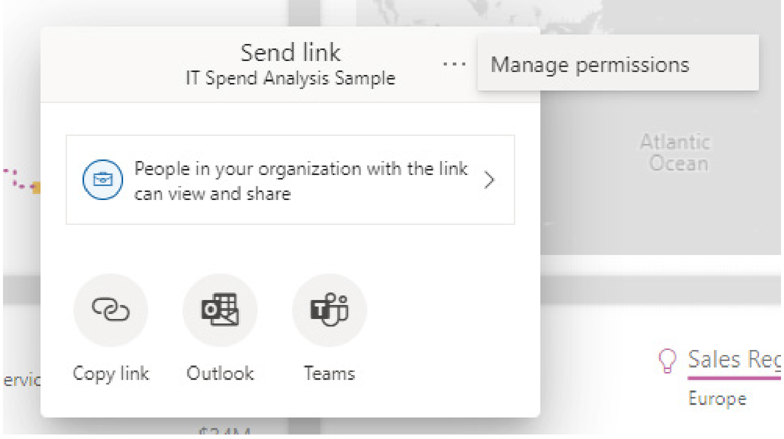
For ‘Specific people’, this type of link allows specific people or groups to access the report. If you select this option, enter the names or email addresses of the people you wish to share with. With this link type you can share to guest users in your organisation’s Azure Active Directory (AAD), but you cannot share to external users who are not guests in your organisation.

Links that give access to ‘People in your organization’ or ‘Specific people’ will always include at least read access. However, you can also specify if you want the link to include or exclude the following permissions as well:

- **Reshare permissions (included by default):** allows recipients to share the report
- **Build permissions (excluded by default):** allows recipients to build content with the data associated with the report.

Links for users with existing access do not have any additional settings because these links do not give any access to the report.

To manage permission and manage links that give access to the report, select the context menu (. . .) in the upper right of the sharing dialog, and then select ‘Manage permissions’:

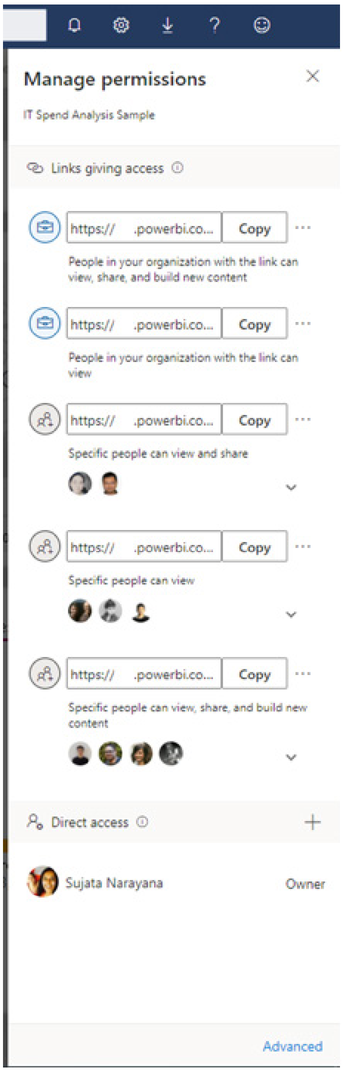


Send link
IT Spend Analysis Sample ··· Manage permissions

People in your organization with the link can view and share

Copy link Outlook Teams

This will launch the ‘Management permissions’ pane, where you can copy or modify existing links or grant users direct access. To modify a given link, click the context menu (. . .).



Manage permissions

IT Spend Analysis Sample

People in your organization with the link can view, share, and build new content

People in your organization with the link can view

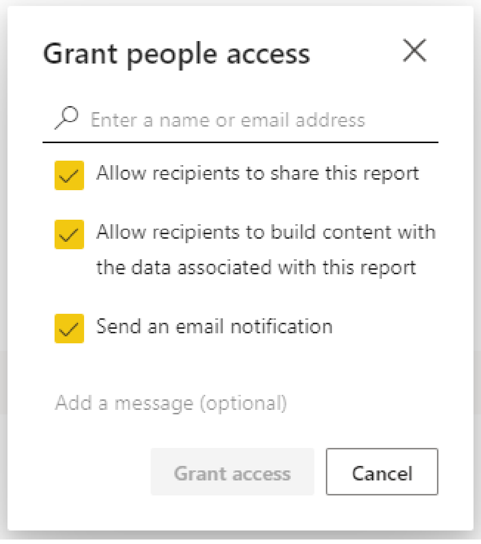
Specific people can view and share

Specific people can view

Specific people can view, share, and build new content

Sujata Narayana Owner

To grant users direct access to the report click the plus icon (+), enter their name or email address, optionally type a message, and click ‘Grant access’.



Grant people access

Enter a name or email address

☑ Allow recipients to share this report

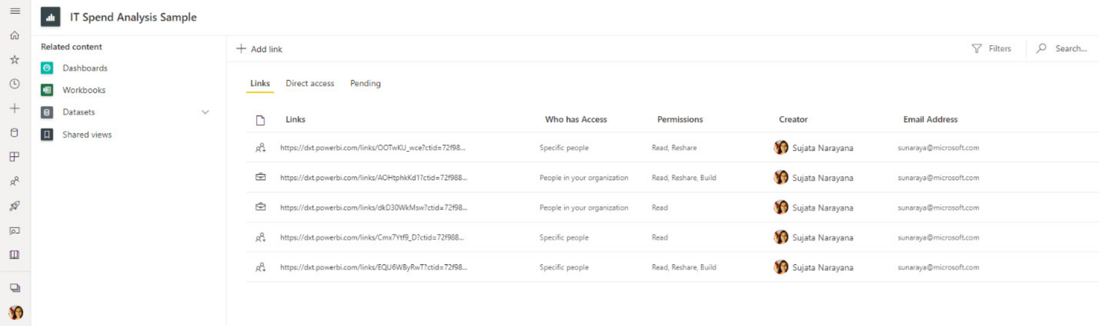
☑ Allow recipients to build content with the data associated with this report

☑ Send an email notification

Add a message (optional)

For additional access management capabilities, select the ‘Advanced option’ in the footer of the ‘Manage permissions’ pane. This will navigate you to the Management page where you can:

- View, manage, and create Links
- View and manage who has Direct access and grant people direct access
- View and manage Pending access requests and invitations
- View and manage Related content
- Apply Filters or Search for specific links or people.

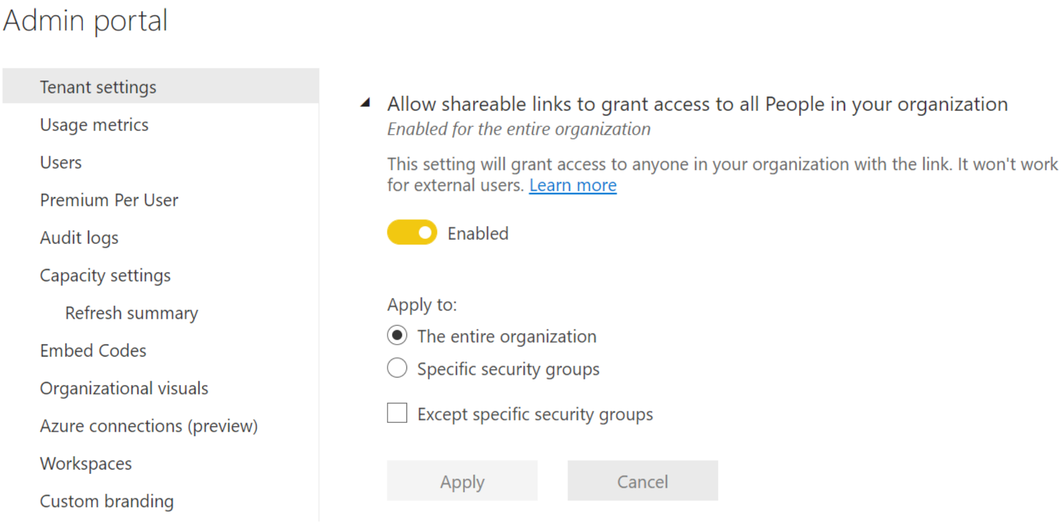


IT Spend Analysis Sample

Links | Direct access | Pending

Links	Who has Access	Permissions	Creator	Email Address
https://dxt.powerbi.com/links/OOTwKU_wce?ctid=72f98...	Specific people	Read, Reshare	Sujata Narayana	sunaraya@microsoft.com
https://dxt.powerbi.com/links/AOHtphkKd1?ctid=72f988...	People in your organization	Read, Reshare, Build	Sujata Narayana	sunaraya@microsoft.com
https://dxt.powerbi.com/links/dkD30WkMsw?ctid=72f98...	People in your organization	Read	Sujata Narayana	sunaraya@microsoft.com
https://dxt.powerbi.com/links/Cmx7Ytf9_D?ctid=72f988...	Specific people	Read	Sujata Narayana	sunaraya@microsoft.com
https://dxt.powerbi.com/links/EQU6WByRwT?ctid=72f98...	Specific people	Read, Reshare, Build	Sujata Narayana	sunaraya@microsoft.com

This update has also added a tenant setting for admins looking to disable creating shareable links to ‘People in your organisation’. You can find this option in the Admin portal by navigating to **Tenant settings -> Export and sharing settings -> Allow shareable links to grant access to all People in your organization:**



Admin portal

Allow shareable links to grant access to all People in your organization

Enabled for the entire organization

This setting will grant access to anyone in your organization with the link. It won't work for external users. Learn more

Enabled

Apply to:

● The entire organization

○ Specific security groups

□ Except specific security groups

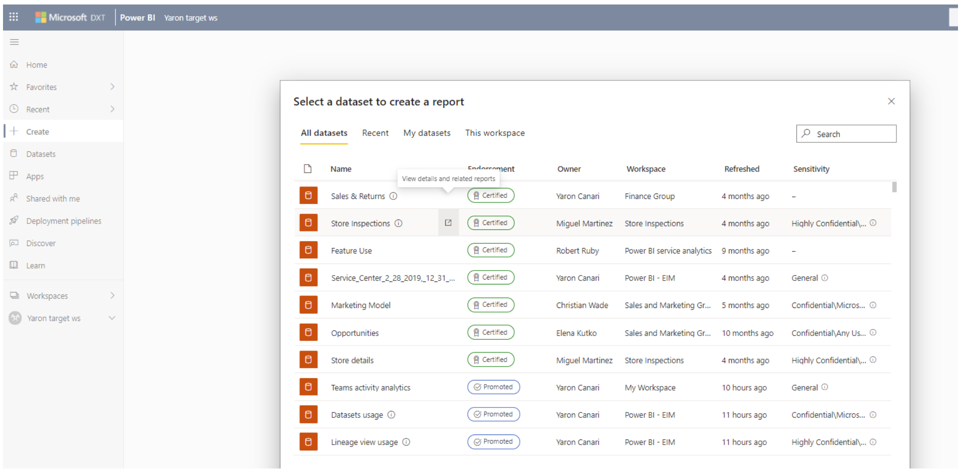
As with other tenant settings, you can enable sharing links to People in your organisation for:

- The entire organisation
- Specific security groups *or*
- Except specific security groups.

If this setting is disabled for a user with share permissions to a report, that user will only be able to share the report via link to ‘Specific people’ or ‘People with existing access’.

Improving the ‘Create on top of published dataset’ experience

When you create a new report in Power BI Service based on top of an existing dataset, you will now see the same datasets hub experience appears in a dialog. You’ll see first in the list datasets that you recently viewed their associated report / dashboard, while still making sure endorsed datasets gets better ranking than non-endorsed ones. You can use the filters to see only datasets that you were recently viewed under ‘Recent’ tab or datasets that you’re the owner of under ‘My datasets’ tab.



Select a dataset to create a report

All datasets | Recent | My datasets | This workspace

Name	Endorsement	Owner	Workspace	Refreshed	Sensitivity
Sales & Returns	Certified	Yaron Canari	Finance Group	4 months ago	–
Store Inspections	Certified	Miguel Martinez	Store Inspections	4 months ago	Highly Confidential\...
Feature Use	Certified	Robert Ruby	Power BI service analytics	9 months ago	–
Service_Center_2_28_2019_12_31_...	Certified	Yaron Canari	Power BI - EIM	4 months ago	General
Marketing Model	Certified	Christian Wade	Sales and Marketing Gr...	5 months ago	Confidential\Micros...
Opportunities	Certified	Elena Kutko	Sales and Marketing Gr...	10 months ago	Confidential\Any Us...
Store details	Certified	Miguel Martinez	Store Inspections	4 months ago	Highly Confidential\...
Teams activity analytics	Promoted	Yaron Canari	My Workspace	10 hours ago	General
Datasets usage	Promoted	Yaron Canari	Power BI - EIM	11 hours ago	Confidential\Micros...
Lineage view usage	Promoted	Yaron Canari	Power BI - EIM	11 hours ago	Highly Confidential\...

New capabilities in Deployment Pipelines

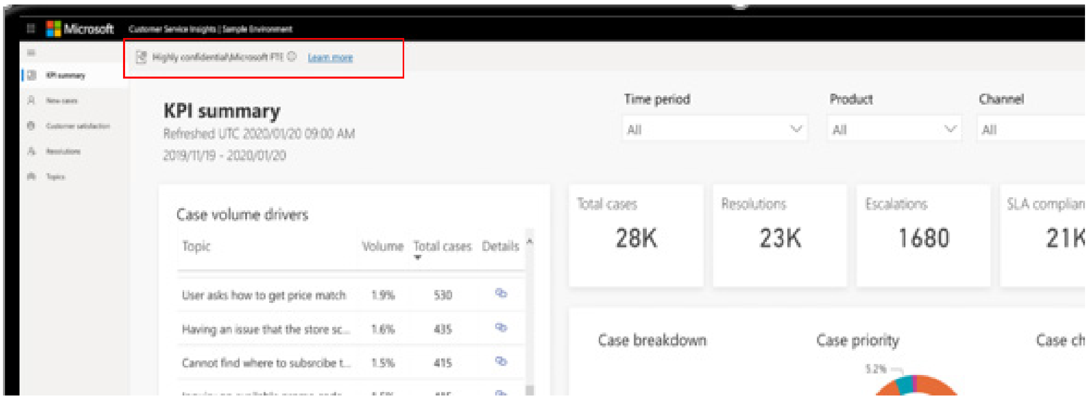
A few capabilities have been added to deployment pipelines, including:

- Paginated reports support
- Data protection integration
- Workspace members can now deploy and update datasets without being the owners.

Enhancements to Sensitivity Labels

SENSITIVITY LABELS IN POWER BI EMBEDDED NOW GENERALLY AVAILABLE

Last year Microsoft announced a public Preview of sensitivity label support in Embedded view. When you embed a visual, report or dashboard that has a sensitivity label applied to it, the sensitivity label will be visible in the Embedded view, and the label and its protection will persist when data is exported to Excel. This capability has now become Generally Available.



Case volume drivers

Topic	Volume	Total cases	Details
User asks how to get price match	1.9%	530	
Having an issue that the store sc...	1.6%	435	
Cannot find where to subscribe t...	1.5%	415	

SECURING THE FULL DATA JOURNEY FROM AZURE TO OFFICE

To ensure that your data remains classified and secured across its “data journey” (to quote Microsoft) from Azure, through Power BI and all the way to Office, there is now a Preview of Power BI MIP (Microsoft Information Protection) label inheritance when importing data from Azure Synapse Analytics and Azure SQL Database.

Power BI datasets connecting to classified data in Azure Synapse Analytics or Azure SQL Database will inherit those labels such that data remains classified and secure when brought into Power BI and inherited when exported to Office. The result is secure, end-to-end sensitivity label inheritance and protection of your business data, from source to point of consumption.

ADMIN API TO SET AND REMOVE LABELS IN POWER BI

To meet compliance requirements, organisations are often required to classify and label all sensitive data in Power BI. This task can be a challenge for tenants that have large volumes of data in Power BI. To make this task easier and more effective, Power BI has introduced new Power BI admin APIs for setting and removing sensitivity labels in Power BI. With these APIs, Power BI administrators can easily and effectively apply sensitivity labels on large number of Power BI artifacts programmatically.

New visuals

The new visuals this month Include:

- Cycle Plot
- graphomate charts
- accoPLANNING
- LegendMap.

Microsoft Charticulator visual

There is a new custom visual in AppSource: the Microsoft Charticulator visual. Previously to use Charticulator, a user would flip between the web app at charticulator.com and Power BI. Now, users can create a wide range of custom chart designs right within Power BI using the new Microsoft Charticulator custom visual.

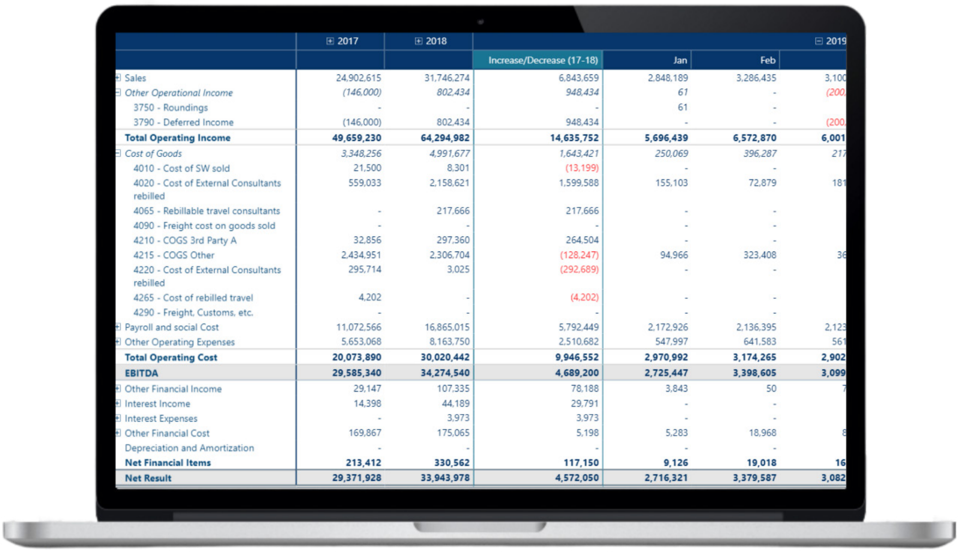
Financial Reporting Matrix by Profitbase

Getting financial statements well presented in Power BI with built-in visuals is quite a challenge, to say the least. Profitbase’s Financial Reporting Matrix enables users to quickly create Excel-like financial statements and reports with tailormade formatting, calculations, expandable / collapsible rows and columns, and more.

In particular, the expandible / collapsible columns have been an oft-requested feature. This functionality has been used in Excel for years, and it’s finally possible in Power BI.

Main features:

- Expand / collapse rows and columns
- Fine-grained control of cell formatting
- Custom columns and row subtotals
- Conditional formatting
- Custom calculations columns
- Formatting and styling.



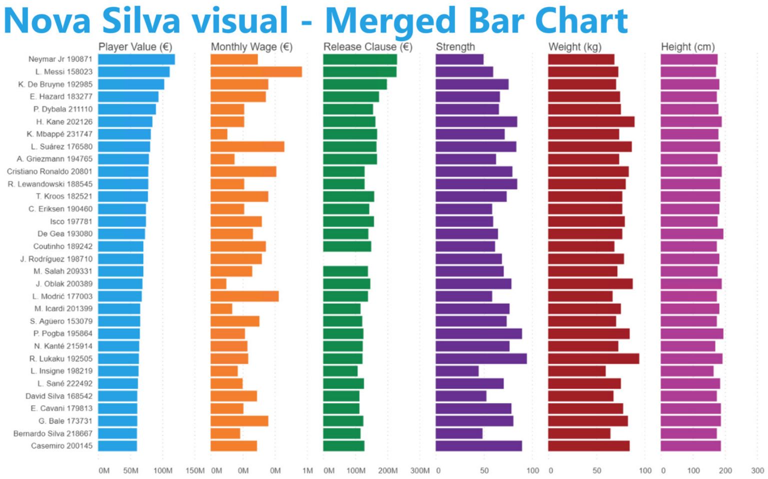
	⊞ 2017	⊞ 2018				⊟ 2019
			Increase/Decrease (17-18)	Jan	Feb	
⊞ Sales	24,902,615	31,746,274	6,843,659	2,848,189	3,286,435	3,100
⊟ *Other Operational Income*	*(146,000)*	*802,434*	*948,434*	*61*	-	*(200,*
3750 - Roundings	-	-	-	61	-	
3790 - Deferred Income	(146,000)	802,434	948,434	-	-	(200,
Total Operating Income	**49,659,230**	**64,294,982**	**14,635,752**	**5,696,439**	**6,572,870**	**6,001**
⊟ *Cost of Goods*	*3,348,256*	*4,991,677*	*1,643,421*	*250,069*	*396,287*	*217*
4010 - Cost of SW sold	21,500	8,301	(13,199)	-	-	
4020 - Cost of External Consultants rebilled	559,033	2,158,621	1,599,588	155,103	72,879	181
4065 - Rebillable travel consultants	-	217,666	217,666	-	-	
4090 - Freight cost on goods sold	-	-	-	-	-	
4210 - COGS 3rd Party A	32,856	297,360	264,504	-	-	
4215 - COGS Other	2,434,951	2,306,704	(128,247)	94,966	323,408	36
4220 - Cost of External Consultants rebilled	295,714	3,025	(292,689)	-	-	
4265 - Cost of rebilled travel	4,202	-	(4,202)	-	-	
4290 - Freight, Customs, etc.	-	-	-	-	-	
⊞ Payroll and social Cost	11,072,566	16,865,015	5,792,449	2,172,926	2,136,395	2,123
⊞ Other Operating Expenses	5,653,068	8,163,750	2,510,682	547,997	641,583	561
Total Operating Cost	**20,073,890**	**30,020,442**	**9,946,552**	**2,970,992**	**3,174,265**	**2,902**
EBITDA	**29,585,340**	**34,274,540**	**4,689,200**	**2,725,447**	**3,398,605**	**3,099**
⊞ Other Financial Income	29,147	107,335	78,188	3,843	50	7
⊞ Interest Income	14,398	44,189	29,791	-	-	
⊞ Interest Expenses	-	3,973	3,973	-	-	
⊞ Other Financial Cost	169,867	175,065	5,198	5,283	18,968	8
Depreciation and Amortization	-	-	-	-	-	
Net Financial Items	**213,412**	**330,562**	**117,150**	**9,126**	**19,018**	**16**
Net Result	**29,371,928**	**33,943,978**	**4,572,050**	**2,716,321**	**3,379,587**	**3,082**

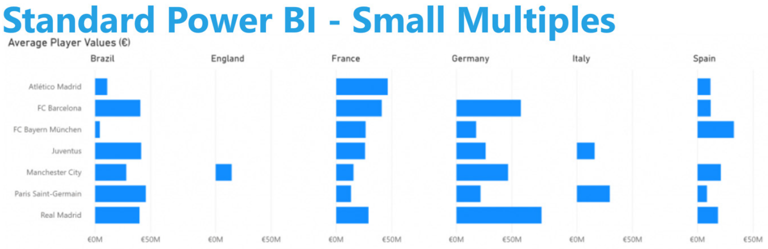
You can drill and summarise along both rows and columns. You can also group and summarise both vertically and horizontally, for example, by date or product categories, without using other programs, such as Excel. Collapsed columns will automatically be summarised – no configuration, modelling or use of other program (*e.g.* Excel) is required.

The Financial Reporting Matrix for Power BI is available from AppSource.

Merged Bar Chart by Nova Silva

The Merged Bar Chart appears to have a lot of similarities with Small Multiples. The key difference is the way these charts allow you to compare values. The Merged Bar Chart focusses on comparing multiple measures (like Player Value, Monthly Wage, Release Clause, *etc*. in the example below) within one specific categorical variable (*e.g*. Soccer Players).





The small multiples focus on segmenting the bars by one or more categorical variables (here, Country of Birth). If you want to compare a single variable over multiple categories, think Small Multiples. However, if you are seeking comparison of multiple independent measures, consider the Merged Bar Chart.

Again, this may be downloaded from the AppSource.

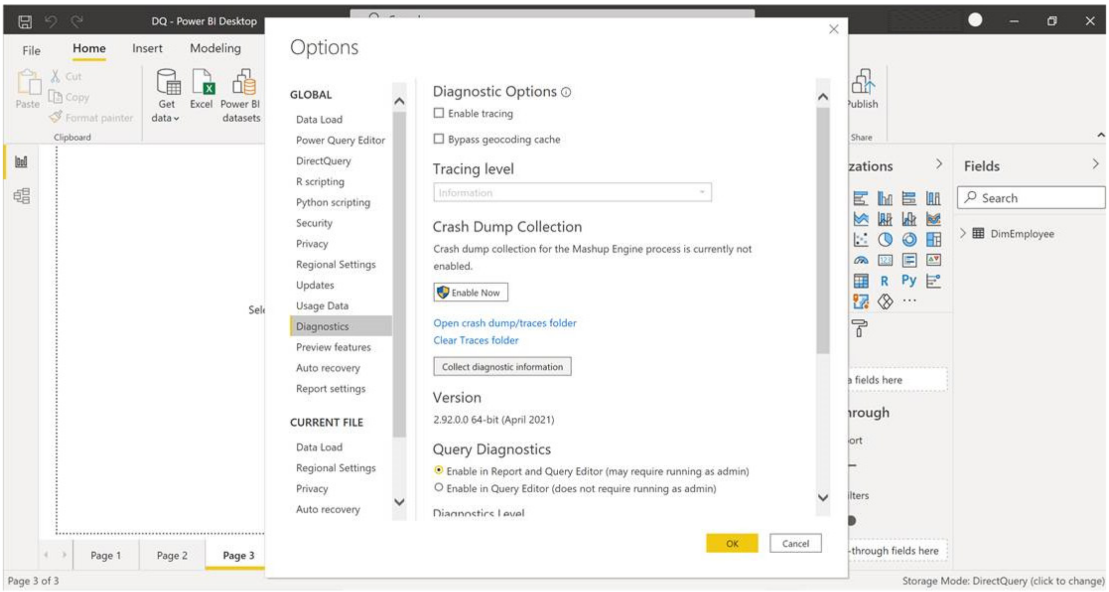
API v3.6.0 is now available

The API v3.6.0 is ready for you to update your visuals. The new API supports empty-dataroles feature, which allows you to receive updates from Power BI without the need to bind any data.

Simplifying collection of diagnostic information

Much of time is often lost in the collection of diagnostic information when you need Microsoft's assistance. This update makes it easier to collect diagnostic information when working with Microsoft's support teams. However, you *should* only use this option when asked to do so by their support team. They should give you clear instructions to follow which will also inform you exactly about which diagnostic information is collected.

You can find the new diagnostic information collection dialog in the Power BI Desktop Options under **Global -> Diagnostics**. In order to capture the maximum amount of diagnostic information, select 'Enable tracing' and then press the 'Collect diagnostic information' button.



Options

Diagnostic Options

☐ Enable tracing

☐ Bypass geocoding cache

Tracing level

Information

Crash Dump Collection

Crash dump collection for the Mashup Engine process is currently not enabled.

Enable Now

Open crash dump/traces folder

Clear Traces folder

Collect diagnostic information

Version

2.92.0.0 64-bit (April 2021)

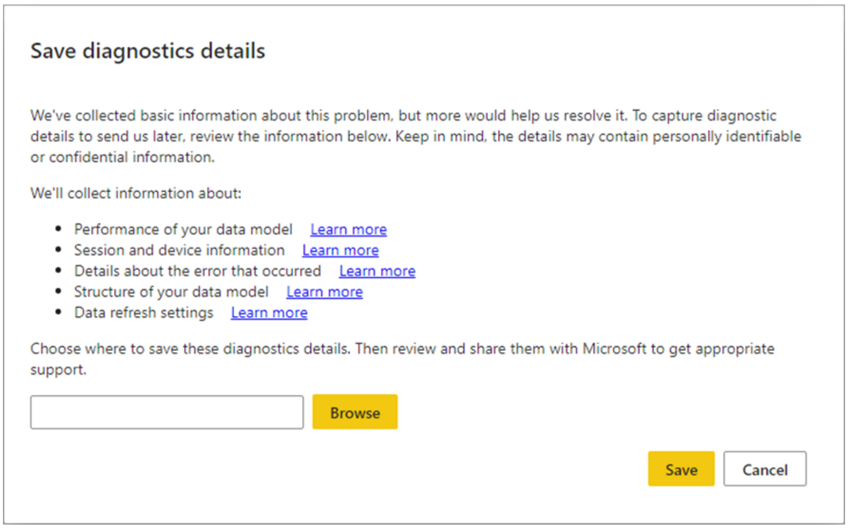
Query Diagnostics

● Enable in Report and Query Editor (may require running as admin)

○ Enable in Query Editor (does not require running as admin)

Diagnostics Level

This opens the Save diagnostics details dialog:



Save diagnostics details

We've collected basic information about this problem, but more would help us resolve it. To capture diagnostic details to send us later, review the information below. Keep in mind, the details may contain personally identifiable or confidential information.

We'll collect information about:

- Performance of your data model Learn more
- Session and device information Learn more
- Details about the error that occurred Learn more
- Structure of your data model Learn more
- Data refresh settings Learn more

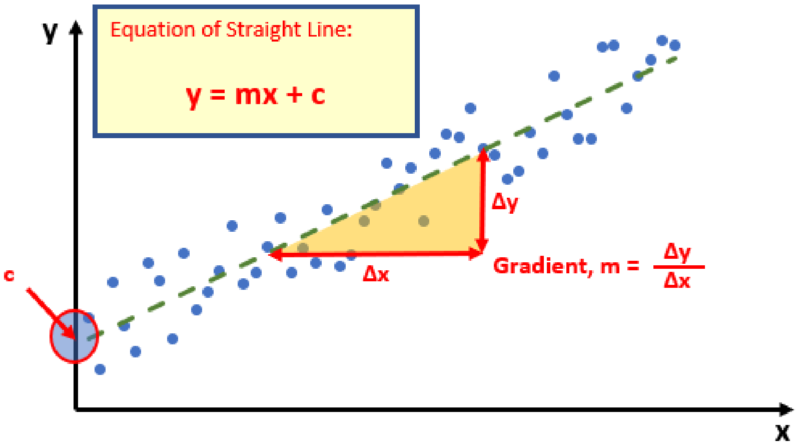
Choose where to save these diagnostics details. Then review and share them with Microsoft to get appropriate support.

You can learn more about the diagnostics collected using the links and choose where you want to save. After you click Save, a ZIP file will be created in the location you specified in the dialog. You can then verify which information is collected before you share it with Microsoft's support team as per their instructions so they may analyse the situation as quickly as possible. They could always stop the errors from occurring in the first place too. ☺

More next month, we're sure!

The A to Z of Excel Functions: FORECAST

A common approach to forecasting is known as *simple linear regression*. This is a technique where historical data is plotted on a chart and a "best straight line" is drawn through the data points to determine a linear relationship, *viz*.



The chart constructed usually is created as a scatter plot, with the independent variable (**x**) on the horizontal axis and the dependent variable (**y**) on the vertical axis. To be clear:

- the **independent variable (x)** is one you either have control of or else may select
- the **dependent variable (y)** is the result of that control or choice, sometimes referred to as **y = f(x)**.

For example, if you are trying to forecast sales, **x** would be the period (*e.g.* months) and **y** would be the forecast / actual amount of sales (*e.g.* dollar amount).

Not all relationships are linear, but a surprising number may be determined by plotting **log x** against **log y** (it does not matter what base is used for the logarithms as long as the same one is chosen). Using this technique to forecast the relationship is known as simple linear regression, which is formally described as a linear approach to modelling the relationship between a scalar response (or dependent variable) and one explanatory variable (or independent variable).

The technique most commonly used to find the "best straight line" is the *ordinary least squares* (OLS) method, where geometrically, the sum of the squared distances between the line and the observed data parallel to the **y**-axis is minimised.

The **FORECAST** function calculates, or predicts, a future value by using existing values. The predicted value is a **y**-value for a given **x**-value. The known values are existing **x**-values and **y**-values, and the new value is predicted by using linear regression.

In Excel 2016, this function has been replaced with **FORECAST.LINEAR** as part of the new set of forecasting functions. It's still available for backward compatibility, but consider using the new function from Excel 2016 / Office 365 onwards.

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

FORECAST(x, known_y's, known_x's)

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

- **x:** this is required and represents the data point for which you wish to predict a value
- **known_y's:** this is required. This is the dependent range of data
- **known_x's:** this is also required. This denotes the independent range of data.

It should be further noted that:

- if x is non-numeric, **FORECAST** returns the *#VALUE!* error value
- if **known_y's** and **known_x's** are empty or contain a different number of data points, **FORECAST** returns the *#N/A* error value
- if the variance of **known_x's** equals zero, then **FORECAST** returns the *#DIV/0!* error value
- the equation for **FORECAST** is **a + bx**, where:

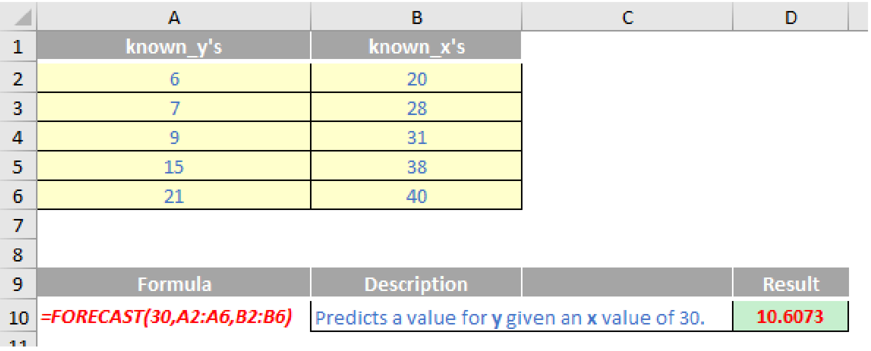
$$a = \bar{y} - b\bar{x}$$

and:

$$b = \frac{\sum(x-\bar{x})(y-\bar{y})}{\sum(x-\bar{x})^2}$$

and where **x bar** and **y bar** are the sample means **AVERAGE(known_x's)** and **AVERAGE(known y's)**.

Please see our example below:

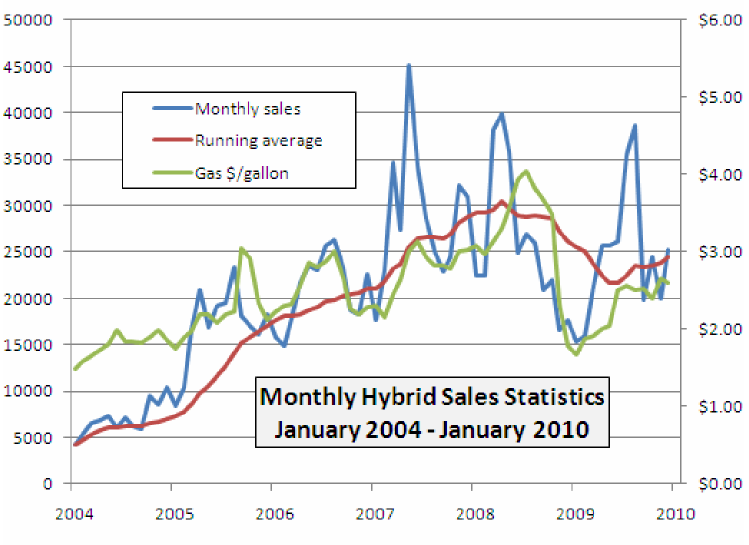


	A	B	C	D
1	known_y's	known_x's		
2	6	20		
3	7	28		
4	9	31		
5	15	38		
6	21	40		
7				
8				
9	Formula	Description		Result
10	*=FORECAST(30,A2:A6,B2:B6)*	Predicts a value for **y** given an **x** value of 30.		10.6073

The A to Z of Excel Functions: FORECAST.ETS

A window function (also known as an apodization or tapering function) is a mathematical function that has a zero value outside of a chosen interval. Uniform distributions and the bell curve are two such window functions commonly used in statistics.

Exponential smoothing is what's known as a rule of thumb technique (*i.e.* not strictly accurate) for smoothing time series data using the exponential window function. The aim is to smooth out historical data to predict trends, *etc*.



The aim is to develop a technique to identify what would be next in a series, i.e. forecast the future. There are various approaches used:

- **Naïve method:** this really does live up to its billing – you simply use the last number in the sequence, *e.g.* the continuation of the series 8, 17, 13, 15, 19, 14, … would be 14, 14, 14, 14, … Hmm, great
- **Simple average:** only a slightly better idea: here, you use the average of the historical series, *e.g.* for the continuation of the series 8, 17, 13, 15, 19, 14, … would be 14.3, 14.3, 14.3, 14.3, …
- **Moving average:** now we start to look at smoothing out the trends by taking the average of the last **n** items. For example, if **n** were 3, then the sequence continuation of 8, 17, 13, 15, 19, 14, … would be 16, 16.3, 15.4, 15.9, 15.9, …
- **Weighted moving average:** the criticism of the moving average is that older periods carry as much weighting as more recent periods, which is often not the case. Therefore, a weighted moving average is a moving average where within the sliding window values are given different weights, typically so that more recent points matter more. For example, instead of selecting a window size, it requires a list of weights (which should add up to 1). As an illustration, if we picked four periods and [0.1, 0.2, 0.3, 0.4] as weights, we would be giving 10%, 20%, 30% and 40% to the last 4 points respectively which would add up to 1 (which is what it would need to do to compute the average).

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

- **Single exponential smoothing:** imagine a weighted average where we consider all of the data points, while assigning exponentially smaller weights as we go back in time. For example, if we started with 0.9, our weights would be (going back in time): 0.9, 0.9^2, 0.9^3, 0.9^4, 0.9^5, … eventually tending to zero.

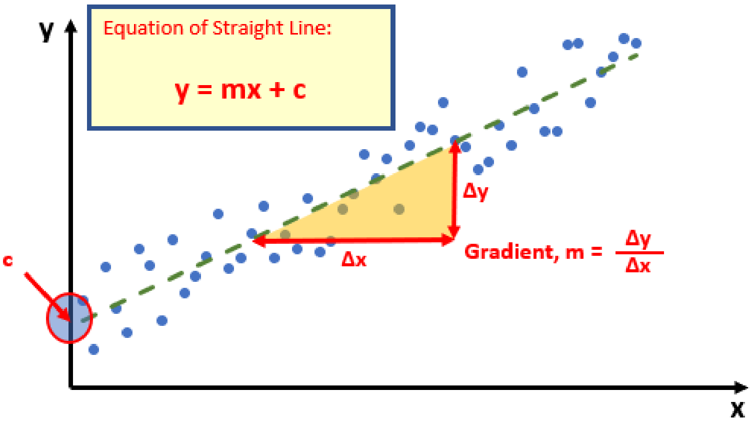
  There is a problem here though: the weights do not add up to 1. The sum of the first three numbers alone is already 2.439. This method needs to be modified using the following succinct and elegant formula:

  $$\hat{y}_x = \alpha \cdot y_x + (1-\alpha) \cdot \hat{y}_{x-1}$$

  This is the sum of two products: **α.y**$_x$ and **(1–α).y^**$_x$. You can think of **α** (alpha) as a sort of a starting weight 0.9 in the above example. It is called the *smoothing factor or smoothing coefficient*. Given essentially we have a weighted moving average with two weights: **α** and **1–α**, the sum of these is 1, so all is acceptable for calculation purposes.

  This gives rise to a recursive technique – hence why this method is called exponential

- **Double exponential smoothing:** all of the methods above are only good for predicting a single point. Double exponential smoothing the **y-intercept** and the **gradient** of two points from a sample, *viz*.



y
Equation of Straight Line:
y = mx + c
c
Δy
Δx
Gradient, m = $\frac{\Delta y}{\Delta x}$
x

Yup, it's that diagram again. We are getting our money's worth, this newsletter.

The **y**-intercept is often known as the *level* and the gradient is known as the *trend*. The trend may be additive (add 10,000 each period) or multiplicative (increase by 10% each period). It is shown in statistics that a ratio (*i.e.* the multiplicative approach) is a more stable predictor.

Double exponential smoothing then is nothing more than exponential smoothing applied to both level and trend. To express this in mathematical notation we now need three equations: one for level, one for the trend and one to combine the level and trend to get the expected **y**.

$$\begin{aligned} \ell_x &= \alpha y_x + (1-\alpha)(\ell_{x-1} + b_{x-1}) && \text{level} \\ b_x &= \beta(\ell_x - \ell_{x-1}) + (1-\beta)b_{x-1} && \text{trend} \\ \hat{y}_{x+1} &= \ell_x + b_x && \text{forecast} \end{aligned}$$

- **Exponential Triple Smoothing (ETS):** this approach takes it to the next level – how to forecast for *many* points. This means we need to consider seasonality too: that a series may be repetitive at regular intervals with **s** seasonal components and length **L**.

  The idea behind triple exponential smoothing is to apply exponential smoothing to the seasonal components in addition to level and trend. The smoothing is applied across seasons, e.g. the seasonal component of the third point into the season would be exponentially smoothed with the one from the third point of last season, third point two seasons ago, *etc*. This, we now have four equations:

$$\begin{aligned} \ell_x &= \alpha(y_x - s_{x-L}) + (1-\alpha)(\ell_{x-1} + b_{x-1}) && \text{level} \\ b_x &= \beta(\ell_x - \ell_{x-1}) + (1-\beta)b_{x-1} && \text{trend} \\ s_x &= \gamma(y_x - \ell_x) + (1-\gamma)s_{x-L} && \text{seasonal} \\ \hat{y}_{x+m} &= \ell_x + mb_x + s_{x-L+1+(m-1)modL} && \text{forecast} \end{aligned}$$

  We now have a third Greek letter, **γ** (gamma) which is the smoothing factor for the seasonal component. The forecast equation now consists of level, trend and the seasonal component.

The **FORECAST.ETS** function calculates or predicts a future value based on existing (historical) values by using this Exponential Triple Smoothing (ETS) algorithm. The predicted value is a continuation of the historical values in the specified target date, which should be a continuation of the timeline. You can use this function to predict future sales, inventory requirements, or consumer trends.

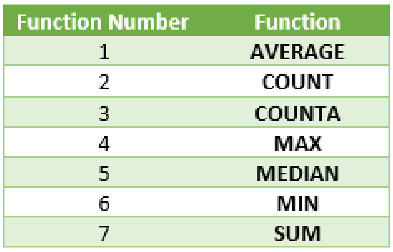
This function requires the timeline to be organised with a constant step between the different points. For example, that could be a monthly timeline with values on the first of every month, a yearly timeline, or a timeline of numerical indices. For this type of timeline, it's very useful to aggregate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

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

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

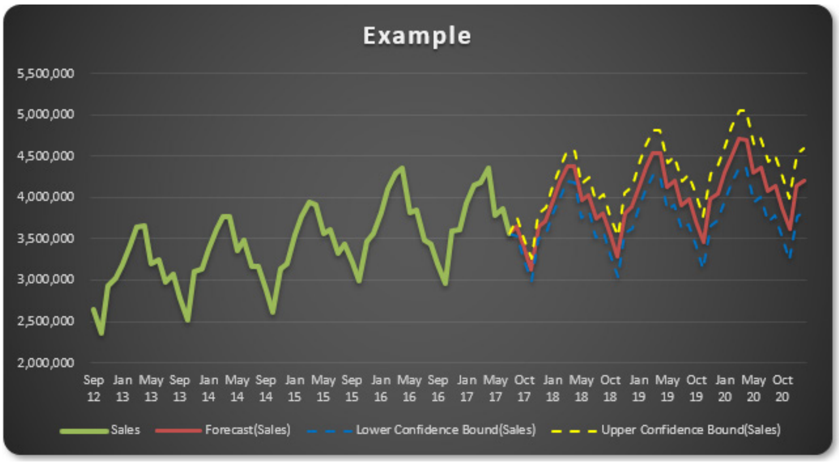
The **FORECAST.ETS** function has the following arguments:

- **target_date:** this is required. This is the data point for which you want to predict a value. The **target_date** may be date / time or numeric. If the target_date is chronologically before the end of the historical timeline, **FORECAST.ETS** returns the *#NUM!* error
- **values:** this is required. The **values** are the historical values, for which you want to forecast the next points
- **timeline:** this is also required. This is the independent array or range of numeric data. The dates in the timeline must have a consistent step between them and cannot be zero (0). The timeline isn't required to be sorted, as **FORECAST.ETS** will sort it implicitly for calculations. If a constant step cannot be identified in the provided timeline, **FORECAST.ETS** will return the *#NUM!* error. If the **timeline** contains duplicate values, **FORECAST.ETS** will return the *#VALUE!* error. If the ranges of the timeline and values are not of the same size, **FORECAST.ETS** will return the *#N/A* error
- **seasonality:** this argument is optional. This is a numeric value with a default value of 1. This means Excel detects seasonality automatically for the forecast and uses positive, whole numbers for the length of the seasonal pattern. 0 indicates no seasonality, meaning the prediction will be linear. Positive whole numbers will indicate to the algorithm to use patterns of this length as the seasonality. For any other value, **FORECAST.ETS** will return the *#NUM!* error
- the maximum supported **seasonality** is 8,760 (number of hours in a year). Any **seasonality** above that number will result in the *#NUM!* error
- **data_completion:** this argument is also optional. Although the **timeline** requires a constant step between data points, **FORECAST.ETS** supports up to 30% missing data, and will automatically adjust for it. Zero (0) will indicate the algorithm to account for missing points as zeros. The default value of 1 will account for missing points by completing them to be the average of the neighboring points
- **aggregation:** this is the final optional argument. Although the **timeline** requires a constant step between data points, **FORECAST.ETS** will **aggregate** multiple points which have the same time stamp. The aggregation parameter is a numeric value indicating which method will be used to aggregate several values with the same time stamp. The default value of 0 will use **AVERAGE**, while other options are **COUNT**, **COUNTA**, **MAX**, **MEDIAN**, **MIN** and **SUM**.



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

As an example:



The A to Z of Excel Functions: FORECAST.ETS.CONFINT

The **FORECAST.ETS.CONFINT** function returns a confidence interval for the forecast value at the specified target date. A confidence interval of 95% means that 95% of future points are expected to fall within this radius from the result **FORECAST.ETS** forecasted (with Normal distribution). Using the confidence interval may help grasp the accuracy of the predicted model. A smaller interval would imply more confidence in the prediction for this specific point.

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

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

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

- **target_date:** this is required. This is the data point for which you want to predict a value. The **target_date** may be date / time or numeric. If the target_date is chronologically before the end of the historical timeline, **FORECAST.ETS.CONFINT** returns the *#NUM!* error
- **values:** this is required. The **values** are the historical values, for which you want to forecast the next points
- **timeline:** this is also required. This is the independent array or range of numeric data. The dates in the timeline must have a consistent step between them and cannot be zero (0). The timeline isn't required to be sorted, as **FORECAST.ETS.CONFINT** will sort it implicitly for calculations. If a constant step cannot be identified in the provided timeline, **FORECAST.ETS.CONFINT** will return the *#NUM!* error. If the timeline contains duplicate values, **FORECAST.ETS.CONFINT** will return the *#VALUE!* error. If the ranges of the timeline and values are not of the same size, **FORECAST.ETS.CONFINT** will return the *#N/A* error
- **confidence_level:** this is optional and represents a numerical value between 0 and 1 (exclusive), indicating a confidence level for the calculated confidence interval. For example, for a 90% confidence interval, a 90% confidence level will be computed (90% of future points are to fall within this radius from prediction). The default value is 95%. For numbers outside of the range (0,1), **FORECAST.ETS.CONFINT** will return the *#NUM!* error
- **seasonality:** this argument is optional. This is a numeric value with a default value of 1. This means Excel detects seasonality automatically for the forecast and uses positive, whole numbers for the length of the seasonal pattern. 0 indicates no seasonality, meaning the prediction will be linear. Positive whole numbers will indicate to the algorithm to use patterns of this length as the seasonality. For any other value, **FORECAST.ETS.CONFINT** will return the *#NUM!* error
- the maximum supported **seasonality** is 8,760 (number of hours in a year). Any **seasonality** above that number will result in the *#NUM!* error
- **data_completion:** this argument is also optional. Although the **timeline** requires a constant step between data points, **FORECAST.ETS.CONFINT** supports up to 30% missing data, and will automatically adjust for it. Zero (0) will indicate the algorithm to account for missing points as zeros. The default value of 1 will account for missing points by completing them to be the average of the neighboring points
- **aggregation:** this is the final optional argument. Although the **timeline** requires a constant step between data points, **FORECAST.ETS.CONFINT** will aggregate multiple points which have the same time stamp. The a**ggregation** parameter is a numeric value indicating which method will be used to aggregate several values with the same time stamp. The default value of 0 will use **AVERAGE**, while other options are **COUNT**, **COUNTA**, **MAX**, **MEDIAN**, **MIN** and **SUM**.

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

The A to Z of Excel Functions: FORECAST.ETS.SEASONALITY

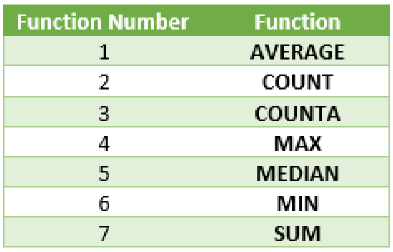
The **FORECAST.ETS.SEASONALITY** function returns the length of the repetitive pattern Excel detects for the specified time series (s). **FORECAST.ETS.SEASONALITY** may be used following **FORECAST.ETS** to identify which automatic seasonality was detected and used in **FORECAST.ETS**. While it can also be used independently of **FORECAST.ETS**, the functions are tied since the seasonality detected in this function is identical to the one used by **FORECAST.ETS**, considering the same input parameters that affect data completion.

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

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

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

- **values:** this is required. The **values** are the historical values, for which you want to forecast the next points
- **timeline:** this is also required. This is the independent array or range of numeric data. The dates in the timeline must have a consistent step between them and cannot be zero (0). The timeline isn't required to be sorted, as **FORECAST.ETS.SEASONALITY** will sort it implicitly for calculations. If a constant step cannot be identified in the provided timeline, **FORECAST.ETS.SEASONALITY** will return the *#NUM!* error. If the timeline contains duplicate values, **FORECAST.ETS.SEASONALITY** will return the *#VALUE!* error. If the ranges of the timeline and values are not of the same size, **FORECAST.ETS.SEASONALITY** will return the *#N/A* error
- **data_completion:** this argument is optional. Although the timeline requires a constant step between data points, **FORECAST.ETS.SEASONALITY** supports up to 30% missing data, and will automatically adjust for it. Zero (0) will indicate the algorithm to account for missing points as zeros. The default value of 1 will account for missing points by completing them to be the average of the neighboring points
- **aggregation:** this is the final optional argument. Although the timeline requires a constant step between data points, **FORECAST.ETS.SEASONALITY** will aggregate multiple points which have the same time stamp. The **aggregation** parameter is a numeric value indicating which method will be used to aggregate several values with the same time stamp. The default value of 0 will use **AVERAGE**, while other options are **COUNT**, **COUNTA**, **MAX**, **MEDIAN**, **MIN** and **SUM**.



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

The A to Z of Excel Functions: FORECAST.ETS.STAT

The **FORECAST.ETS.STAT** function returns a statistical value as a result of time series forecasting. The statistic type is determined by this function too (see below).

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

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

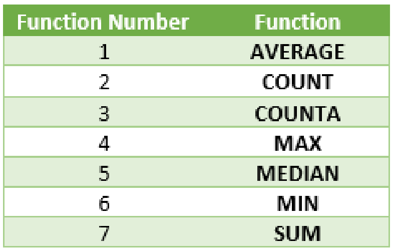
The **FORECAST.ETS.STAT** function has the following arguments:

- **values:** this is required. The values are the historical values, for which you want to forecast the next points
- **timeline:** this is also required. This is the independent array or range of numeric data. The dates in the timeline must have a consistent step between them and cannot be zero (0). The timeline isn't required to be sorted, as **FORECAST.ETS.STAT** will sort it implicitly for calculations. If a constant step cannot be identified in the provided timeline, **FORECAST.ETS.STAT** will return the *#NUM!* error. If the timeline contains duplicate values, **FORECAST.ETS.STAT** will return the *#VALUE!* error. If the ranges of the timeline and values are not of the same size, **FORECAST.ETS.STAT** will return the *#N/A* error
- **statistic_type:** this is required. This is a numeric value between 1 and 8, indicating which statistic will be returned for the calculated forecast:

Function Number	Statistic	Description
0	Alpha parameter of ETS algorithm	Returns the base value parameter; a higher value gives more weight to recent data points
1	Beta parameter of ETS algorithm	Returns the trend value parameter; a higher value gives more weight to the recent trend
3	Gamma parameter of ETS algorithm	Returns the seasonality value parameter; a higher value gives more weight to the recent seasonal period
4	MASE metric	Returns the mean absolute scaled error metric; this is a measure of the accuracy of the forecasts

Function Number	Statistic	Description
5	SMAPE metric	Returns the symmetric mean absolute percentage error metric: this is an accuracy measure based on percentage errors
6	MAE metric	Returns the symmetric mean absolute percentage error metric: this is an accuracy measure based on percentage errors
7	RMSE metric	Returns the root mean squared error metric: this is a measure of the differences between predicted and observed values
8	Step size detected	Returns the step size detected in the historical timeline.

- **seasonality:** this argument is optional. This is a numeric value with a default value of 1. This means Excel detects seasonality automatically for the forecast and uses positive, whole numbers for the length of the seasonal pattern. 0 indicates no seasonality, meaning the prediction will be linear. Positive whole numbers will indicate to the algorithm to use patterns of this length as the seasonality. For any other value, **FORECAST.ETS.STAT** will return the *#NUM*! error
- The maximum supported **seasonality** is 8,760 (number of hours in a year). Any seasonality above that number will result in the *#NUM!* error
- **data_completion:** this argument is also optional. Although the **timeline** requires a constant step between data points, **FORECAST.ETS.STAT** supports up to 30% missing data, and will automatically adjust for it. Zero (0) will indicate the algorithm to account for missing points as zeros. The default value of 1 will account for missing points by completing them to be the average of the neighboring points
- **aggregation:** this is the final optional argument. Although the timeline requires a constant step between data points, **FORECAST.ETS.STAT** will aggregate multiple points which have the same time stamp. The aggregation parameter is a numeric value indicating which method will be used to aggregate several values with the same time stamp. The default value of 0 will use AVERAGE, while other options are **COUNT**, **COUNTA**, **MAX**, **MEDIAN**, **MIN** and **SUM**.



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

The A to Z of Excel Functions: FORECAST.LINEAR

The **FORECAST.LINEAR** function calculates, or predicts, a future value by using existing values. The predicted value is a **y**-value for a given **x**-value. The known values are existing **x**-values and **y**-values, and the new value is predicted by using linear regression.

This function is new to Excel 2016, and replaces the legacy **FORECAST** function as part of the new set of forecasting functions. **FORECAST** is still available for backward compatibility, but consider using the new function from Excel 2016 / Office 365 onwards.

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

FORECAST.LINEAR(x, known_y's, known_x's)

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

- **x:** this is required and represents the data point for which you wish to predict a value
- **known_y's:** this is required. This is the dependent range of data
- **known_x's:** this is also required. This denotes the independent range of data.

It should be further noted that:

- if **x** is non-numeric, **FORECAST.LINEAR** returns the *#VALUE!* error value
- if **known_y's** and **known_x's** are empty or contain a different number of data points, **FORECAST.LINEAR** returns the *#N/A* error value
- if the variance of **known_x's** equals zero, then **FORECAST.LINEAR** returns the *#DIV/0!* error value
- the equation for **FORECAST.LINEAR** is **a + bx**, where:

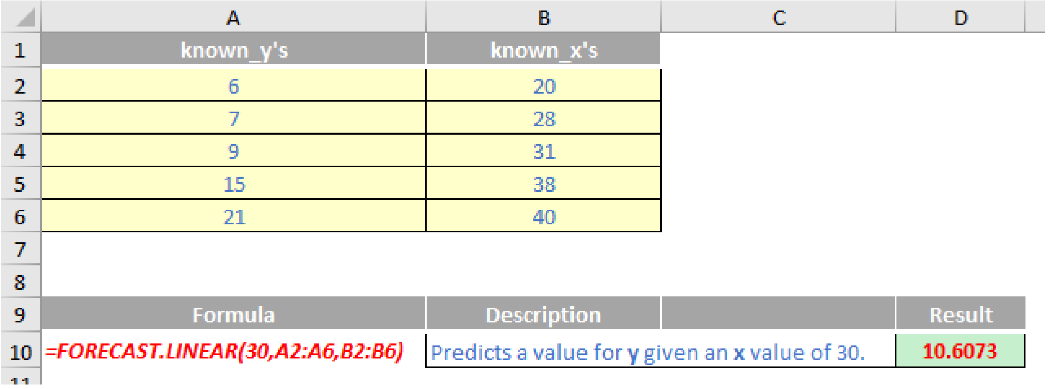
$$a = \bar{y} - b\bar{x}$$

and:

$$b = \frac{\sum (x-\bar{x})(y-\bar{y})}{\sum (x-\bar{x})^2}$$

and where **x bar** and **y bar** are the sample means **AVERAGE(known_x's)** and **AVERAGE(known y's)**.

Please see our final example below:

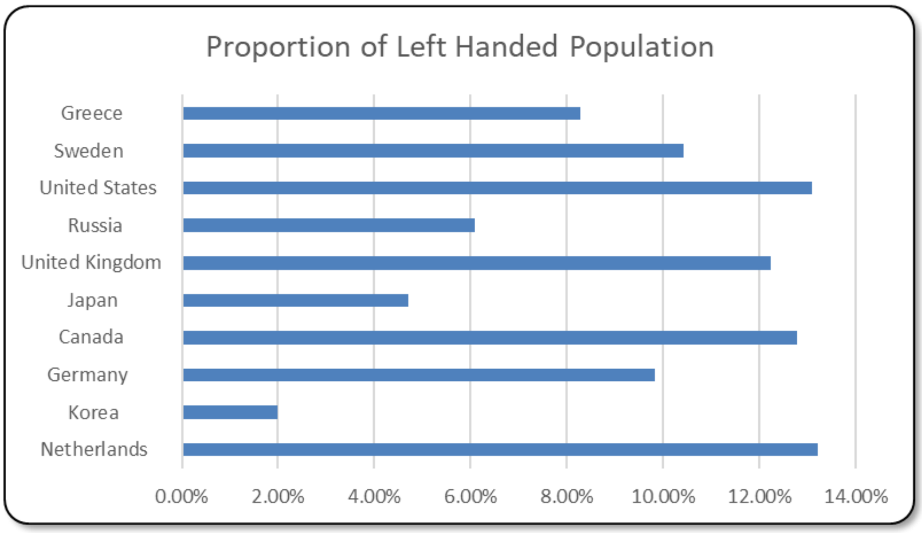


	A	B	C	D
1	known_y's	known_x's		
2	6	20		
3	7	28		
4	9	31		
5	15	38		
6	21	40		
7				
8				
9	Formula	Description		Result
10	=FORECAST.LINEAR(30,A2:A6,B2:B6)	Predicts a value for y given an x value of 30.		10.6073

More Excel Functions next month.

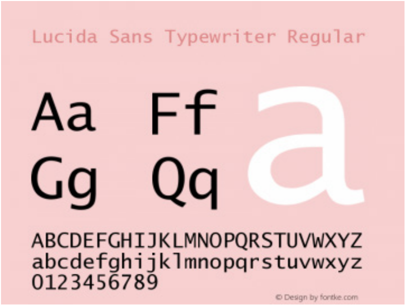
Beat the Boredom Suggested Solution

This month's challenge was about thinking shifting text alignment to the left for the vertical text labels of a bar chart - which default to the right.



The hack used to align the text to the left is using monospace type fonts. Monospace type fonts are fonts where letters and characters each occupy the same amount of horizontal space. Think of the good old typewriter which uses monospace fonts and old-fashioned typeset print media. There are two monospaced fonts that come default with Windows / Office: Courier New and Lucida Sans Typewriter.

Courier New Regular
Aa Ff
Gg Qq
a
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
0123456789
© Design by fontke.com

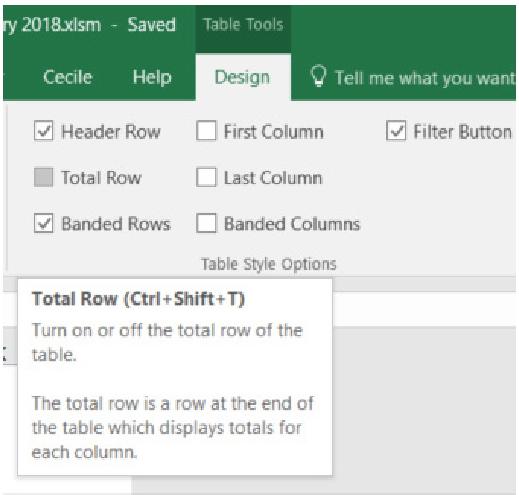

Lucida Sans Typewriter Regular
Aa Ff
Gg Qq
a
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
0123456789
© Design by fontke.com

What this means is that spaces can be appended to the text to generate spaces as filler within the label to give the illusion of being left aligned.

How many spaces would be needed to be added to the end? Well, let's look at the **longest** text label and use that as our base. The number of characters of a table can be found with the **LEN** function. Notice how the data is stored in a table. Let's add a column to the right with the following formula:

=LEN([@Country])

In order to find the maximum, let's put in a totals row. The total row can be added by using the 'Table Style Options' subgroup of Design tab under 'Table Tools' in the Ribbon:



☑ Header Row □ First Column ☑ Filter Button

□ Total Row □ Last Column

☑ Banded Rows □ Banded Columns

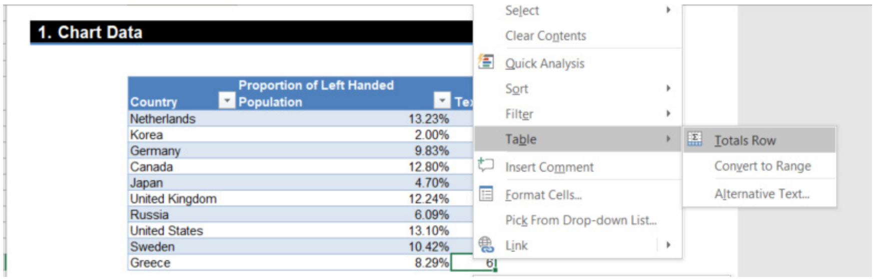
Table Style Options

Total Row (Ctrl+Shift+T)

Turn on or off the total row of the table.

The total row is a row at the end of the table which displays totals for each column.

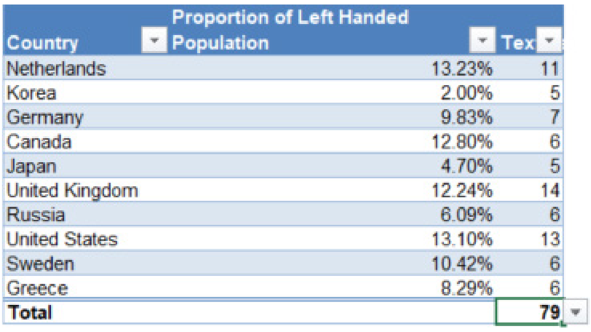
or alternatively in the contextual menu upon right clicking a cell in the table:



1. Chart Data

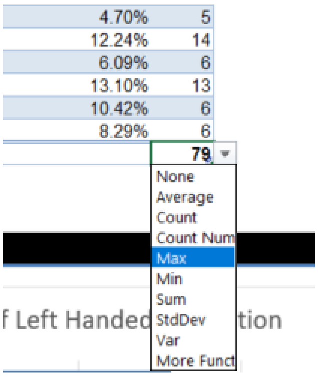
Country	Proportion of Left Handed Population	Te
Netherlands	13.23%	
Korea	2.00%	
Germany	9.83%	
Canada	12.80%	
Japan	4.70%	
United Kingdom	12.24%	
Russia	6.09%	
United States	13.10%	
Sweden	10.42%	
Greece	8.29%	6

That will make the Totals Row – but this calculates the **SUM** of the Table:



Country	Proportion of Left Handed Population	Tex
Netherlands	13.23%	11
Korea	2.00%	5
Germany	9.83%	7
Canada	12.80%	6
Japan	4.70%	5
United Kingdom	12.24%	14
Russia	6.09%	6
United States	13.10%	13
Sweden	10.42%	6
Greece	8.29%	6
Total		**79**

Notice that there is a drop-down menu on right of the **Total** cell.

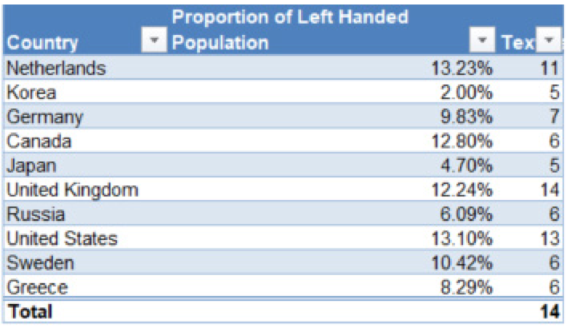


4.70%	5
12.24%	14
6.09%	6
13.10%	13
10.42%	6
8.29%	6
	79

From there, **MAX** may be selected and it will change the formula in the bar to be:

=SUBTOTAL(104,[Text Length])

Don't forget to change the row label to "**Max**" instead of "**Total**"! Now, the table looks like this:

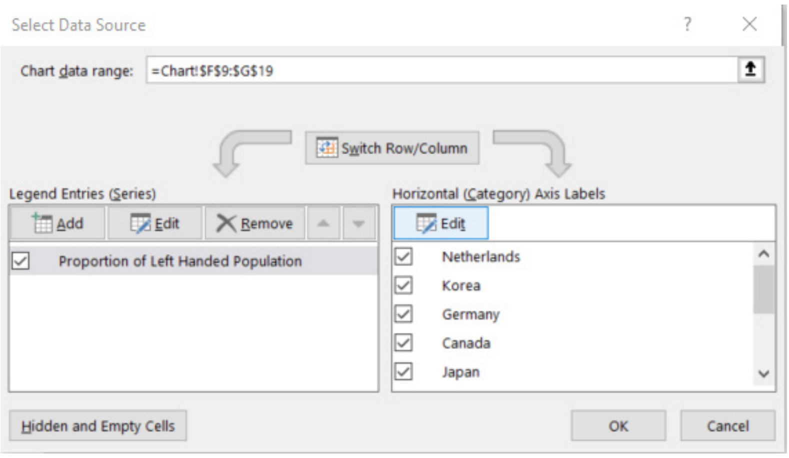


Country	Proportion of Left Handed Population	Tex
Netherlands	13.23%	11
Korea	2.00%	5
Germany	9.83%	7
Canada	12.80%	6
Japan	4.70%	5
United Kingdom	12.24%	14
Russia	6.09%	6
United States	13.10%	13
Sweden	10.42%	6
Greece	8.29%	6
Total		**14**

So the number of spaces required to fill in the rest of the label would be the maximum of the length LESS the actual length of the current label. The spaces can then be generated using the **REPT** function with the following formula:

=[@Country]&REPT(" ",Tbl_LeftHandedness[[#Totals],[Text Length]]-[@[Text Length]])

The next step is to point the labels of the chart to the new column. Go to 'Select Data Source' (found on the right click menu of the chart or the Data subgroup of the Design tab under the Chart Tools tab in the Ribbon).



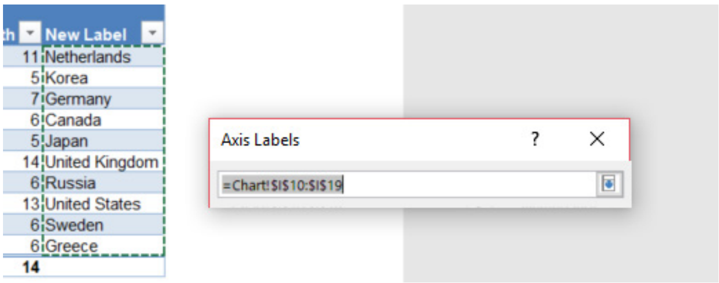
Select Data Source

Chart data range: =Chart!$F$9:$G$19

Legend Entries (Series): ☑ Proportion of Left Handed Population

Horizontal (Category) Axis Labels: ☑ Netherlands ☑ Korea ☑ Germany ☑ Canada ☑ Japan

Click the Edit button under 'Horizontal (Category) Axis Labels' and select the **New Label** column.

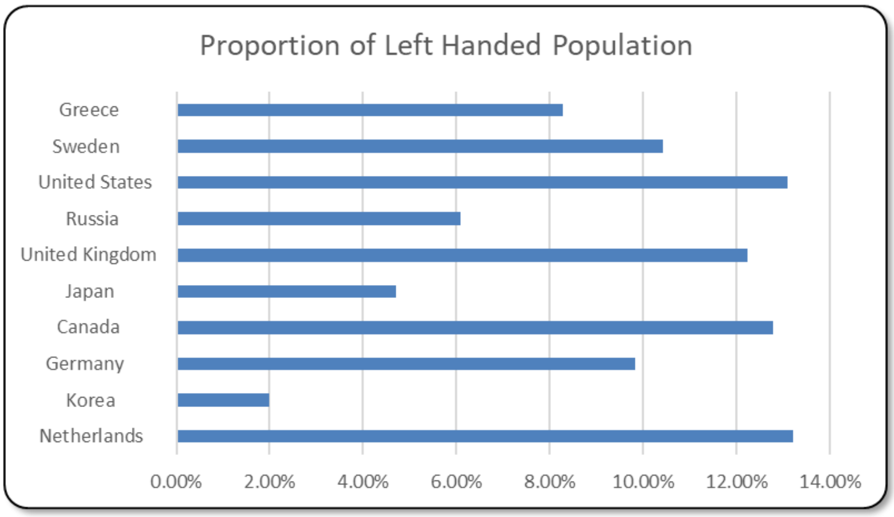


th	New Label
11	Netherlands
5	Korea
7	Germany
6	Canada
5	Japan
14	United Kingdom
6	Russia
13	United States
6	Sweden
6	Greece
14	

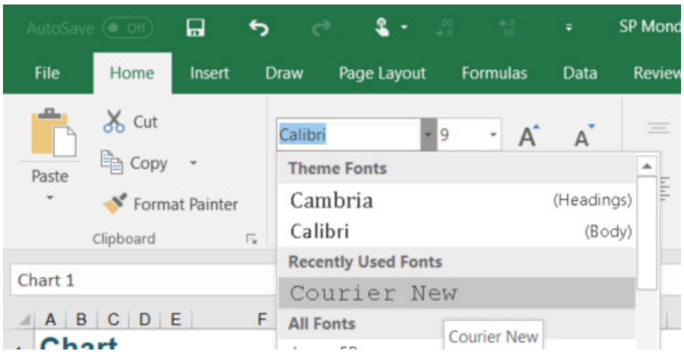
Axis Labels

=Chart!$I$10:$I$19

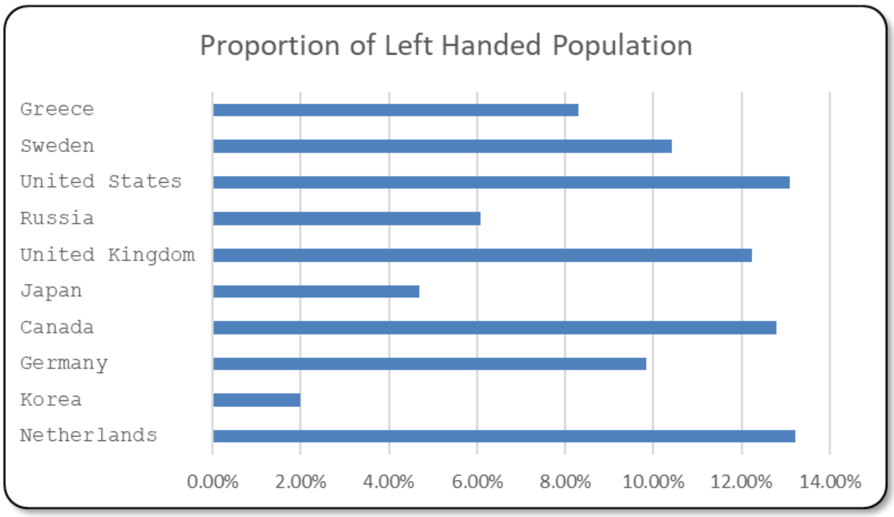
The chart will now be updated. Let's see how it looks.



Notice how the labels appear strangely offset? That's because the font is not spaced equally between each letter! Change the font to one of the monospaced fonts above by clicking on the axis labels and selecting it in the Font subgroup of the Home tab in the Ribbon.



The chart will now be updated. Let's see how it looks.



Hope that tickled the left side of the brain!

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	10 - 12 May 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	17 May 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	1 Day
Online (Australia)	Financial Modelling	18 - 19 May 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	15 - 17 Jun 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	22 Jun 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	1 Day
Online (Australia)	Financial Modelling	23 - 24 Jun 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	19 - 21 Jul 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	26 Jul 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	1 Day
Online (Australia)	Financial Modelling	27 - 28 Jul 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	23 - 25 Aug 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	3 Days

Location	Course	Date	Date	Duration	Duration
Online (Australia)	Excel Tips and Tricks	30 Aug 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	1 Day
Online (Australia)	Financial Modelling	31 Aug - 1 Sep 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	29 Sep - 1 Oct 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	6 Oct 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	1 Day
Online (Australia)	Financial Modelling	7 - 8 Oct 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	3 - 5 Nov 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	10 Nov 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	1 Day
Online (Australia)	Financial Modelling	11 - 12 Nov 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	2 Days
Online (Australia)	Power Pivot, Power Query and Power BI	8 - 10 Dec 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	3 Days
Online (Australia)	Excel Tips and Tricks	15 Dec 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 GMT	1 Day
Online (Australia)	Financial Modelling	16 - 17 Dec 2021	09:00-17:00 AEDT	(-1 day) 22:00-07:00 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 May, let's look at the **F5** tips:

Keystroke	What it does
F5	Go To / Refresh File List
ALT + F5	Refresh
CTRL + F5	Restore window
SHIFT + F5	Find dialog
CTRL + ALT + F5	Refresh all

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