



## Happy New Year!

With 2020 vision (*get it?*), have you made any New Year's Resolutions? What about working more efficiently with your data as you turn it into information, knowledge and finally, wisdom. We discuss (*below*).

And if you are not too stuffed from seeing out the old year, we have got the band back together to see in the new year with all the usual hit singles and deep cuts (VBA, Power Pivot, Power Query, Power BI and for the modellers, three functions, **DOLLAR**, **DOLLARDE** and **DOLLARFR**). I hope some of it makes sweet music for you.

Do you have 2020 foresight?

Liam Bastick, Managing Director, SumProduct

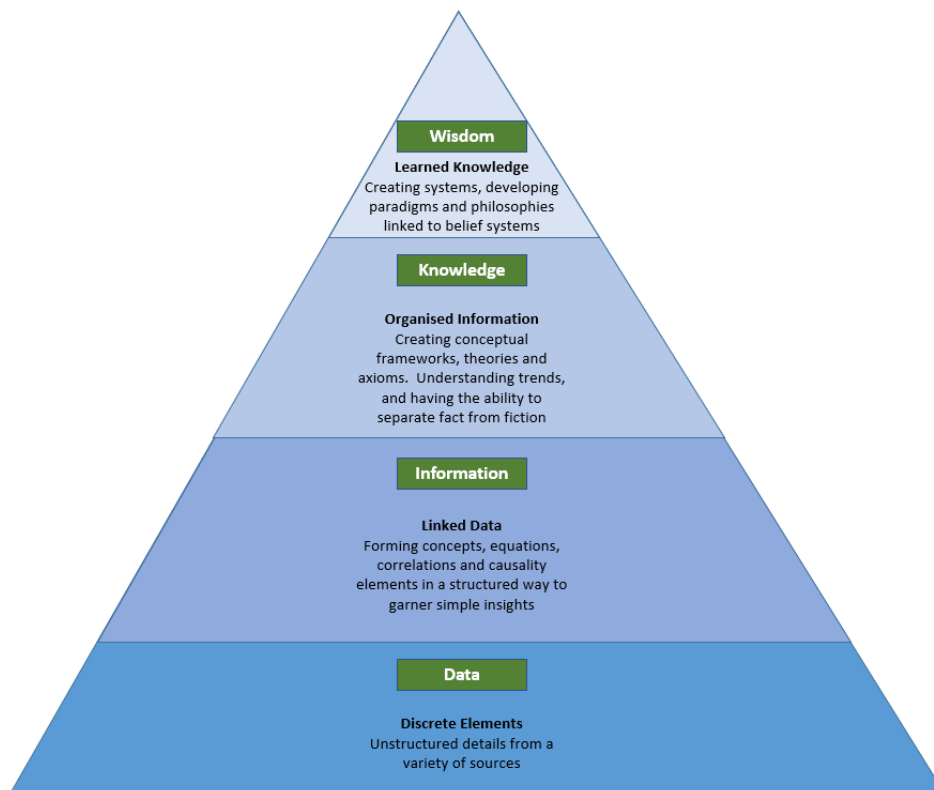


## Are You Using Business Intelligence Intelligently?

In today's competitive environment, the need for timely data which can be sliced and diced in a myriad of ways has never been more vital. Senior managers need access to sales and costs reports immediately to identify trends, recognise issues and allow their companies to steal a march on their rivals. And yet many organisations attempt to collate this with the humble spreadsheet – a reliable, “safe” solution but not one that takes

data by the scruff of the neck and wring out knowledge in an efficient and effective way. That method's *old school*.

The paradigm has shifted and dynamic change is required. Not just from businesses but from service providers too. Over 150 software companies have jumped on the information highway bandwagon to turn data into something, well, *more*.



Past spreadsheets and management information systems typically served up a succession of knowledge metrics, namely Key Performance Indicators (KPIs), that were relevant 18 months previously when the process was specified / decided, with results 21 days old or worse. Knowledge and wisdom based upon these methods were mere pipedreams.

With a lack of agility, flexibility and timeliness, these mechanisms made “business intelligence” the epitomal oxymoron. The speed of change had to change speedily.

Many companies have tried to plug this expectation gap. Some have tried to take the organisation along the whole journey, from Data to Wisdom, whereas others have focused on key areas such as informative charts and dashboards (Information). There's nothing wrong with seeking a complete solution or a simple fix – as long as the end user recognises what the tools address and just as importantly, how expensive they are. That's right: you have to analyse the analytical tools.

You need to realise the main objectives of Business Intelligence (BI) tools:

- **Extraction:** data is pulled from one or more sources (Data)
- **Transformation:** data is manipulated and / or cleaned so that it may be analysed more efficiently (Information)
- **Loading:** data is added to a reporting system (Information)
- **Calculation, relation and correlation:** data is linked as relationships are defined and calculations are performed to gain Knowledge
- **Reporting and what-if analysis:** outputs are prepared numerically or graphically with the ability to perform sensitivity and scenario analyses (Knowledge and Wisdom).

When assessing such revolutionary tools, here's 10 important questions to consider:

1. **Can this solution access my business data?** Not everything is in Excel; no matter how sophisticated and clever the tool is if it cannot forge links to the required data sources then it is not right for you
2. **How easy is it to work with live data?** Foreign exchange traders will not accept monthly updating. An extreme example of course, but the point is well made: how easy is it to update data and how long does that process take?
3. **How easy is it to create ad hoc reports?** Even the best spec'd software requires the ability for customisation. KPIs move on and as information leads to knowledge, wisdom will require changes to reporting over time and other dimensions. Can new data be added from new sources with relative ease? Can granularity of data be modified?
4. **Does it provide automated insights?** Today's analyst needs to analyse rather than prepare. Tomorrow's analyst will seek help in finding correlations in large data sets which may be too unwieldy for the less experienced or time poor
5. **Is it user friendly?** Access is much more powerful than everyone's second favourite software, Excel – but it's not used because it is more sophisticated and unforgiving of mistakes. Software requiring PhDs in Information Technology are unlikely to lead the market or be a term of endearment for analysts
6. **What's the bang for buck?** What does it deliver per element of scarce resource? This may not be a dollar figure: it may be reporting time, staff hours, computing requirements, bandwidth, etc.
7. **What is the level of customer support?** Is it easy to get answers to questions quickly? Do answers make sense and come from a helpdesk you trust?
8. **How easy is it to share the outputs / workload / reports?** Some tools require logging in to a secure system which prevents customers, shareholders and other key stakeholders from accessing information that may be relevant. Can reporting be shared inter- and intra-departmentally?
9. **Does the software require internet access?** Many businesses do not trust the non-fluffy Cloud or have strict IT / governmental regimes where placing data on third party servers is prohibited. This is often a deal-breaker
10. **How secure is the data?** Some systems are more secure than others. These systems tend to be less flexible and more expensive, so a trade-off must be considered.

It is difficult to determine what is right for you, but there are the "usual suspects". I have deliberately put these in alphabetical order (i.e. no ranking is explicitly or implicitly implied) to help you on your journey to BI enlightenment:

Software	Description
IBM	Building on from the more "traditional" Cognos, Watson has become part-celebrity in this environment and has been a case study for many studying accountants
Microsoft Power BI	A relatively new kid on the block, the software giant has made no secret of the fact it wants to cut out the middle-man and has set its price and ease of use accordingly. The joke is often told that the only three buttons any management information system requires are "OK", "Cancel" and "Export to Excel" becomes more relevant by the day
Oracle	Favoured by business's using its software already, it is recognised for having powerful BI and visualisation tools
Qlik	Offering a suite of tools (QlikView is particularly well known), this software is competitively priced and focused on self-service business intelligence

Software	Description
Salesforce	Well known for its Customer Relationship Management (CRM) system, Salesforce offers a range of BI tools such as Einstein Analytics, particularly well-suited to its CRM endeavours
SAP	Many organisations use the Enterprise Resource Planning (ERP) tool, so SAP has developed several BI tools to work in tandem such as BusinessObjects and SAP Analytics Cloud
SAS	Specialist in this area, SAS provides various tools that provide visual reporting solutions (Visual Analytics) to powerful tools more often the realm of data scientists than necessarily data analysts
Tableau	Also highly regarded, Tableau is one of the original entrants in the BI arena, offering one of the more intuitive and versatile visualisation tools, with real-time and predictive analytics. Having been procured recently by Salesforce, it will be interesting to see what this popular software will bring in 2020.

It can be daunting to survey the options, but it's riskier to continue to rely on spreadsheeting. Yes, here at SumProduct, we tend to focus on the Power BI suite of tools (because we are business consultants and training in this area), but you should choose *something*.

Manage your data now before it manages your organisation. Make that your New Year's resolution for 2020!

## Do Your Models Need a Health Check?

January tends to be a quieter month for many and a time for reflection. Perhaps you should consider your financial models. Numerous studies have shown that most financial models contain errors. While the precise estimates based on samples vary (some studies report numbers as high

as 100% of models with over 150 formulae), it's very clear that human error, poor spreadsheet practice and a lack of training all contribute in various ways to errors in financial models.



Let's go over some of the statistics:

- An Ernst & Young study (2006) of over 400 spreadsheets demonstrated that every spreadsheet contained an error of some kind
- A KPMG study in 1998 of 22 spreadsheets showed that 91% contained significant errors
- Lawrence and Lee (2004) looked at 30 spreadsheet audits that showed that 100% contained errors, with a cell error rate of 6.9% (i.e. one error every 17 unique formulae)
- Experienced staff conducting peer reviews of spreadsheets generally only identify 40-60% of errors when working individually. When self-reviewing, this falls to less than 20%.

These are pretty damning numbers! Unfortunately, they match up with our experience of reviewing client models. This applies to the small and medium enterprise (SME) business building a budget model as equally as it does to investment banking firms trying to finance the next public private partnership (PPP) transaction. When we've reviewed models, we find that there is generally an error or potential error for every 10 to 30 unique formulae that we review. Some of these errors might be very minor and not have any effect on the final numbers - units or labels might be missing,

a calculation might not work properly given the inputs, but coincidentally gets the right number (often zero). However, there are plenty of cases where an error has had a significant impact on the bottom-line profitability forecast, or the valuation of a project. We've seen valuations that have been out to the tune of a billion dollars, and profit forecasts of millions turn into losses of the same. Companies go into administration and eventual receivership after an error in the forecasts have been identified, sending entire business models down the drain.

Even with this borne in mind, there seems like there is little incentive for many to get a model reviewed. As an unsecured creditor, perhaps it's even worse incentive to be a model reviewer! Seriously though, it's far more important to be aware of issues while they're just forecasts and before they become realities, when there's still time to plan and make changes to the business model.

So, with that in mind, what sort of model review do you need? There are a few different types:

- **Model Audit:** This is perhaps the most stringent of the types we can consider. An auditor will generally review all of the formulae in a model by check that they're calculating things and being used appropriately; they will use software to check that formulae are consistent across rows and down tables; they will look at the outputs of a model to check that they're reasonable and appropriate. At the end of the process, once all issues are satisfactorily responded to, the auditor will generally release an opinion letter that can be relied upon by stakeholders (usually negative assurance, i.e. nothing adverse has come to our attention that hasn't been rectified)
- **Model Review:** Similar to a Model Audit, but without the opinion letter at the end of the process. This is usually just as rigorous, but since the audit firm doesn't provide reliance at the end of process, this is generally cheaper
- **Agreed Procedures Review:** This is not typically as rigorous as a Model Audit, generally because the review is being conducted to a fixed budget or targeted list of tasks. The goal is not to complete a comprehensive review, but rather, to focus on a few specific risk areas to obtain the best "bang for buck".

Usually, the process will involve inspecting more than one version (iteration) of the model – a model will go back and forth between the company and the auditor, as errors are identified, fixed, and sent back to be reviewed again. On average, for a reasonably large spreadsheet a Model Review or Model Audit might take between three and six iterations to identify and finalise all of the issues.

While we don't tend to go on about this area of expertise too much, in addition to our modelling and training services, we also provide model audit and other review services. If you have a spreadsheet (or multiple!) that you need to rely on for business, and you want some assurance that your spreadsheets are reasonably robust, why not give us a call? Drop us a line at [contact@sumproduct.com](mailto:contact@sumproduct.com) – we work all over the world!

## Visual Basics

*We thought we'd run an elementary series going through the rudiments of Visual Basic for Applications (VBA) as a springboard for newer users. This month, we discuss making your macros look more professional.*

From last month, we're going to finish off our discussion on recording macros by showing you how to clean them up and make them look more professional. Let's consider the code that we finished off with last time:

```
Sub Macro1 ()  
'  
' Macro1 Macro  
'  
'  
  
Range("Start").Select  
Selection.Copy  
Range("End").Select  
ActiveSheet.Paste  
Range("Old").Select  
Application.CutCopyMode = False  
Selection.Copy  
Range("New").Select  
ActiveSheet.Paste  
  
End Sub
```

Let's start by looking at the green lines. The four lines at the top that start with an inverted comma (') are **comments**. In technical terms, any row that starts with an inverted comma is ignored by a macro, so you can type whatever you like in these cells. Therefore, practically speaking, they are used by people to leave notes and remarks about the macro for others to review in the future.

```
'  
' This macro copies two starting values to two new target cells.  
'
```

Next, let's look at each of the lines of "real" code. The first line selects the "Start" range name. The second line then takes whatever had been selected and copies it. Wouldn't it make more sense to combine them instead?

`Range("Start").Select`  
`Selection.Copy`            `Range("Start").Copy`

Hence, instead of selecting a specific cell, then copying the selection, we can instruct Excel to simply copy the specific cell in a single step. In case you're wondering, this doesn't work for the Paste command, since **CTRL + V** isn't something you do to a specific cell (think what happens if you

copy a picture, rather than another cell) but instead, something you do to an *entire sheet*. However, if you use Paste Special, you can apply that action to the Range object.

Another tell-tale sign that a macro has been recorded is in the indentation for the code – typically indentation is only used for loops and If statements. We'll get to those in due course. For now, though, you can delete the indents at the start of each line.

```
Sub CopyAndPaste ()  
' This macro copies two starting values to two new target cells.  
  
Range("Start").Copy  
Range("End").Select  
ActiveSheet.Paste  
Application.CutCopyMode = False  
Range("Old").Copy  
Range("New").Select  
ActiveSheet.Paste  
  
End Sub
```

There we go – a more professional looking macro that doesn't immediately look like it's been recorded.

More next month.

## Power Pivot Principles

We continue our series on the Excel COM add-in, Power Pivot. This month, we discuss the proper practices we should adopt when creating measures.

Before we create more complicated measures, it is good practice to convert each field that we are going to use into a measure. There's a good reason to do this, even if it seems like an unnecessary chore. In Excel, if you add / remove rows or columns Excel's formulae will update automatically. However, if you change a field name in Power Pivot, the dependent measures will not update and will not calculate as a consequence.

Therefore, in order to avoid changing many measures when a field is renamed, it is simpler to create an aggregate measure of that field first and then have all further measures refer to that intermediate measure.

Finally, the last bit that we can easily change is the very first lines – what's the name of the macro? Rather than call it Macro1, we can call it something a bit more meaningful.

Therefore, this will save us a lot of time if we turn fields into 'simple' measures first.

Converting fields into measures first yields two benefits:

1. It saves time in the long run; we avoid having to use an aggregation function each time we reference the field
2. It simplifies the formula when creating complex measures.

Let's create a measure for the 'Product Cost'. In this example, we have completed the measure dialog box with the following details:

The screenshot shows the 'Measure' dialog box with the following details:

- Table name: Sales
- Measure name: Product Cost
- Description: (empty)
- Formula: `=Sum(Sales[TotalProductCost])`
- Formatting Options:
  - Category: Currency
  - Symbol: \$
  - Decimal places: 2
  - Use 1000 separator (.):

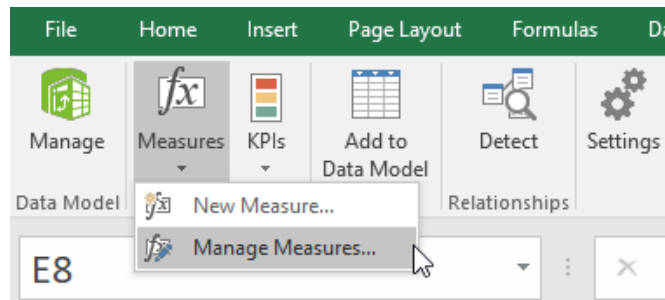
Next up, we'll construct a similar 'Sales' measure:

The screenshot shows the 'Measure' dialog box with the following details:

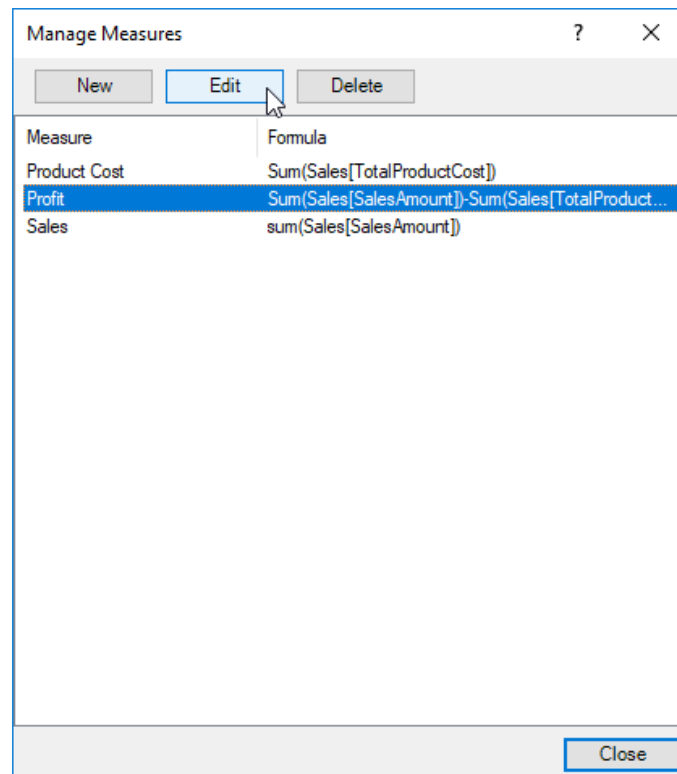
- Table name: Sales
- Measure name: Sales
- Description: (empty)
- Formula: `=sum(Sales[SalesAmount])`
- Formatting Options:
  - Category: Currency
  - Symbol: \$
  - Decimal places: 2
  - Use 1000 separator (.):



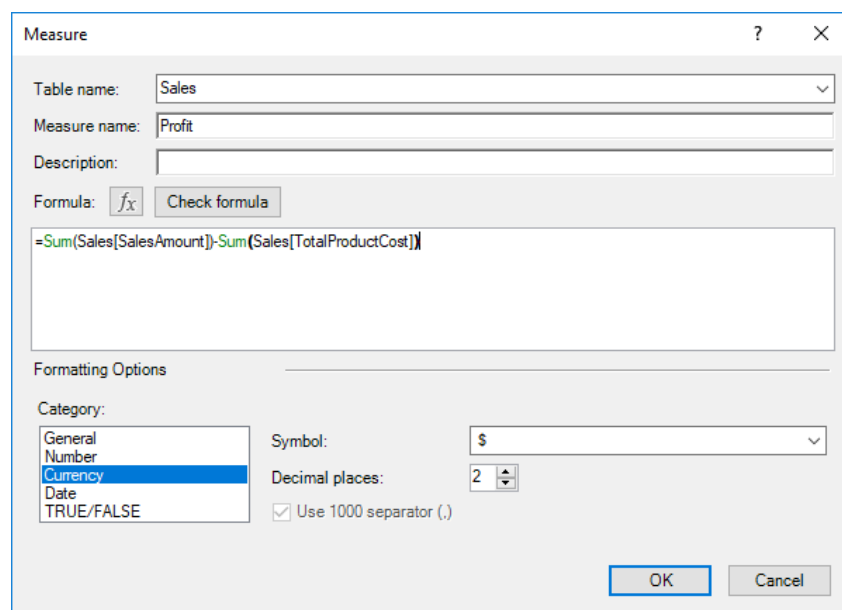
We can now move on to modify the 'Profit' measure. From the 'Measures' drop-down menu, select 'Manage Measures...':



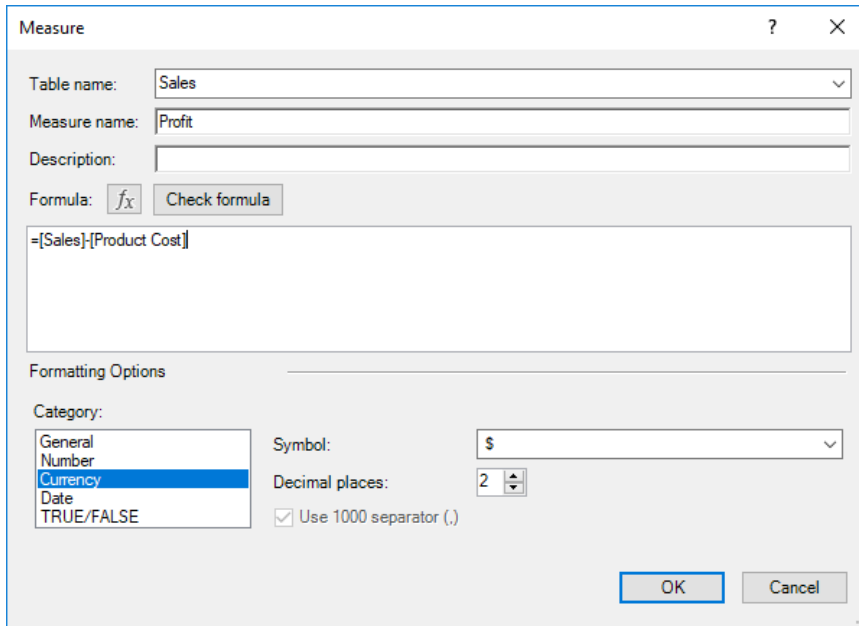
The 'Manage Measures' dialog box appears displaying all measures along with their formulae. Select the Profit measure and then press 'Edit' button:



The measures dialog box appears allowing us to edit the formula.



We can now condense the formula down to:



See how complicate the new formula is? Simple!

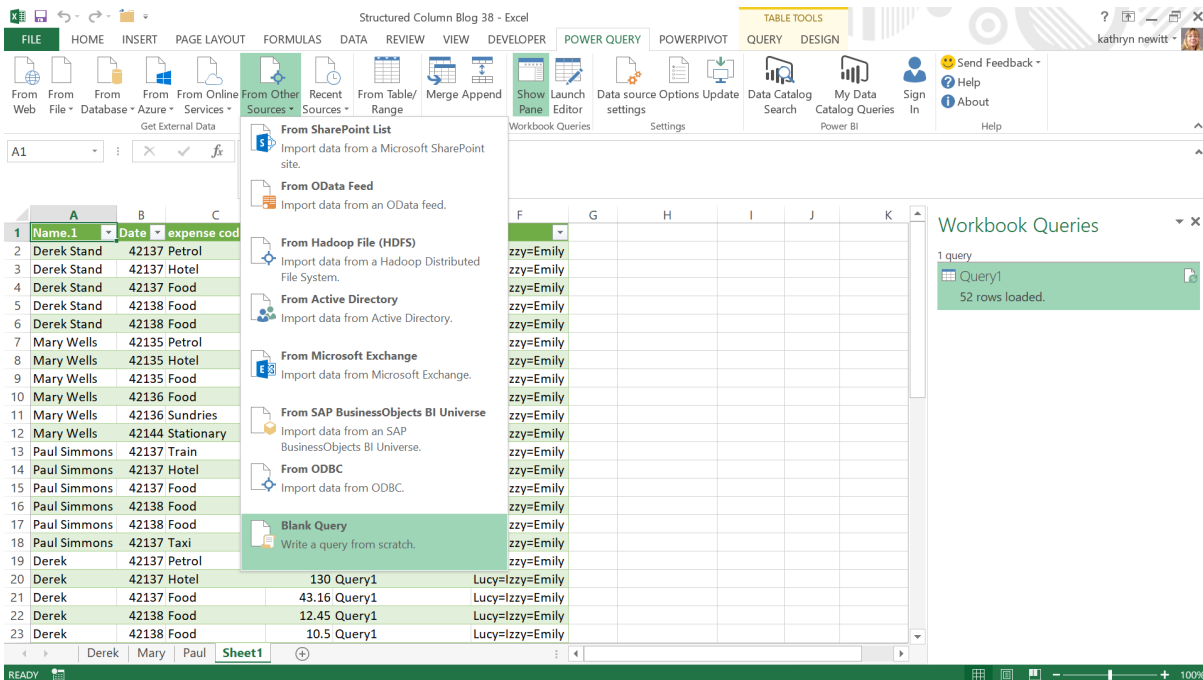
More next month...

## 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](http://www.sumproduct.com/blog). If you wish to read more in the meantime, simply check out our Blog section each Wednesday. This month, we take a look at an easy way to view and test M code functions against data in a workbook.

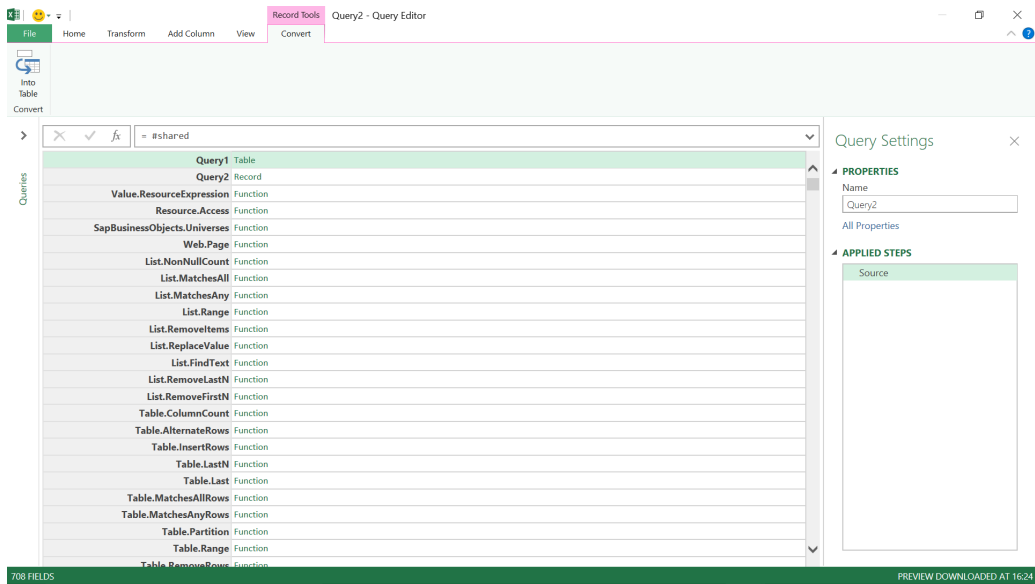
Previously, we have described how to get to a description of the functions that can be used in M code to manipulate data by accessing the Microsoft help pages. However, there is another way to get to the functions. Let's show you.

We'll begin with a blank query:

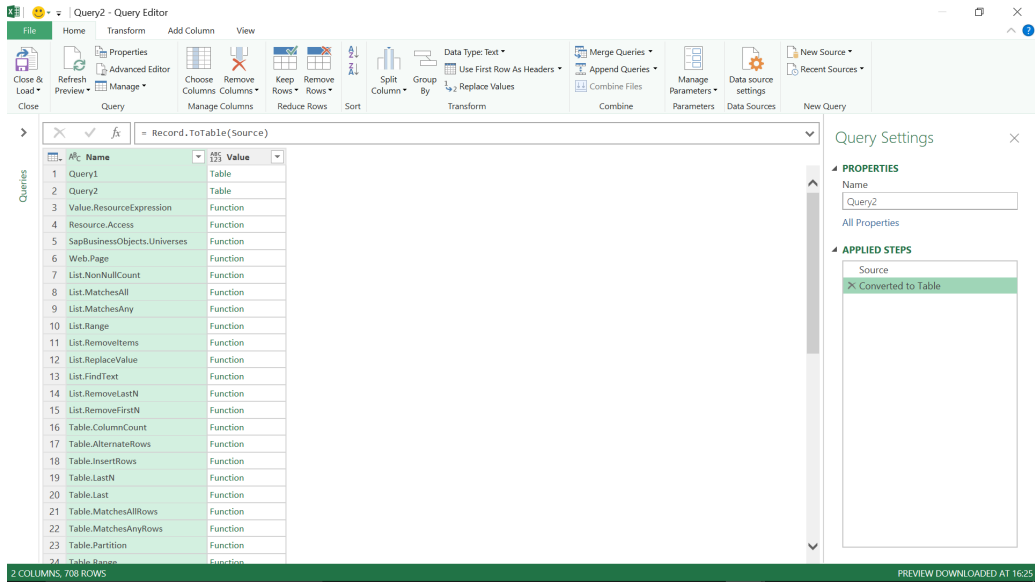


In the formula bar, enter the formula

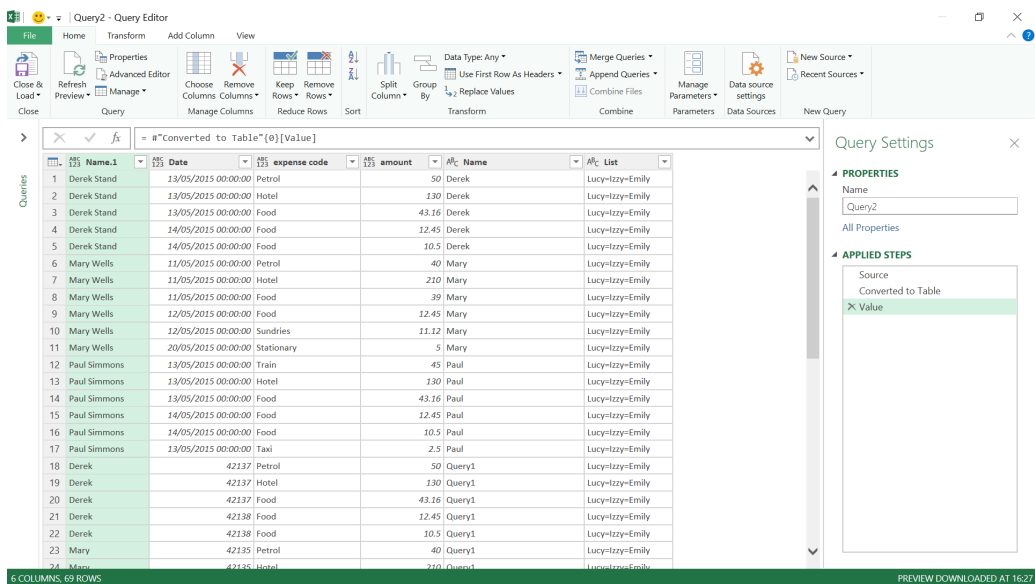
`=#shared`



The screen displays a list of elements in our workbook, and a list of functions available to us. In order to investigate further, choose the option at the top left of the screen in the 'Convert' section, namely 'Into Table' (feel free to select something else if you can find it!):

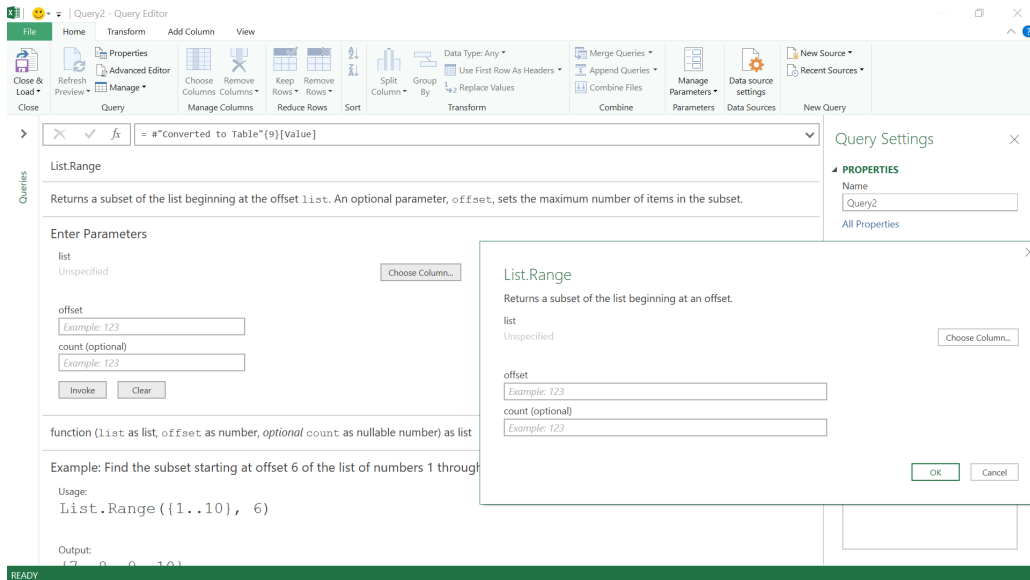


The top rows show the queries in our workbook, and if we click on 'Table' next to one of them you can see the data in the query:

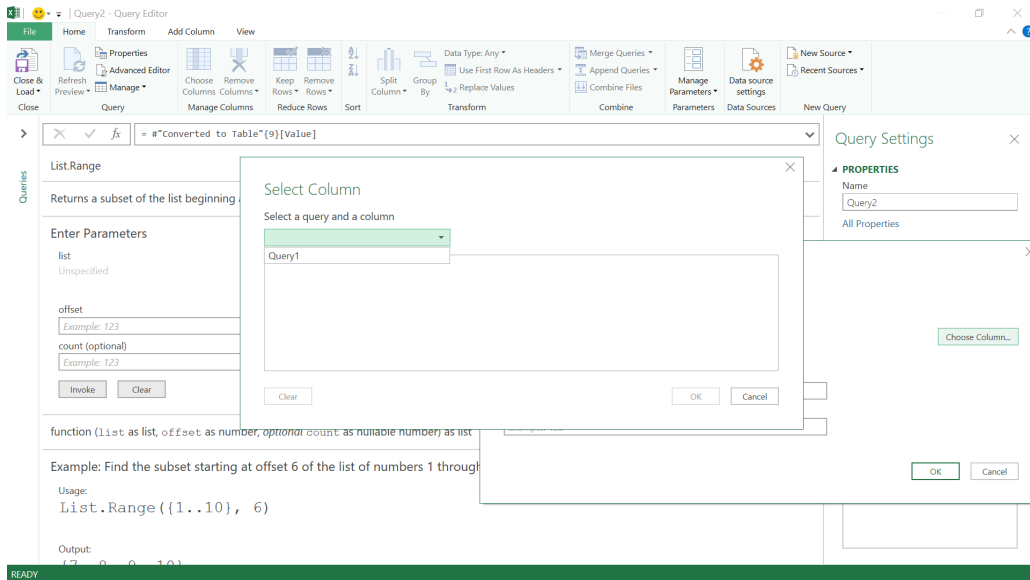




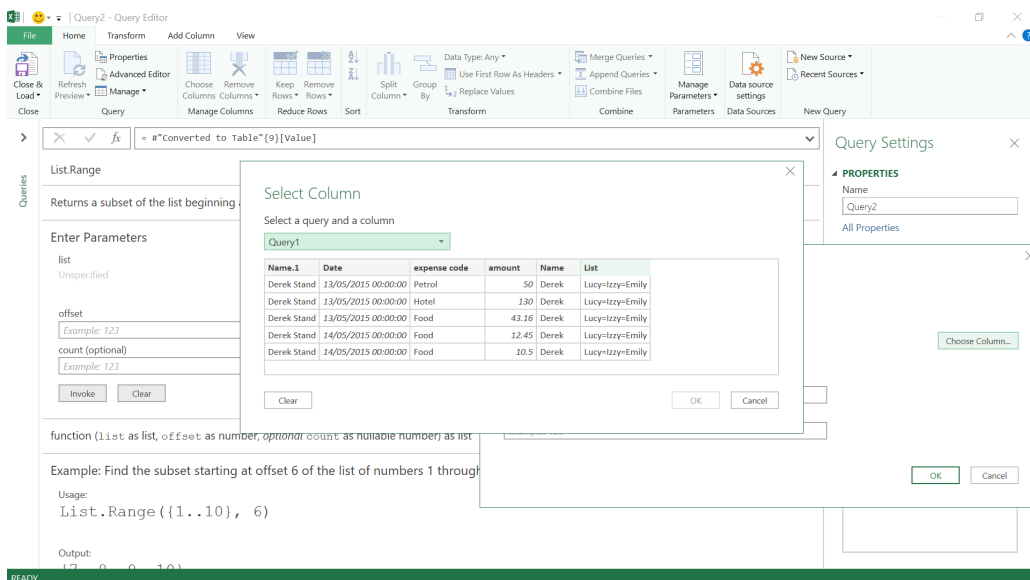
Going back to my list of functions, further down the rows you can see the **List.Range** function we used previously for creating running totals. If we try clicking on the 'Function' label next to it:



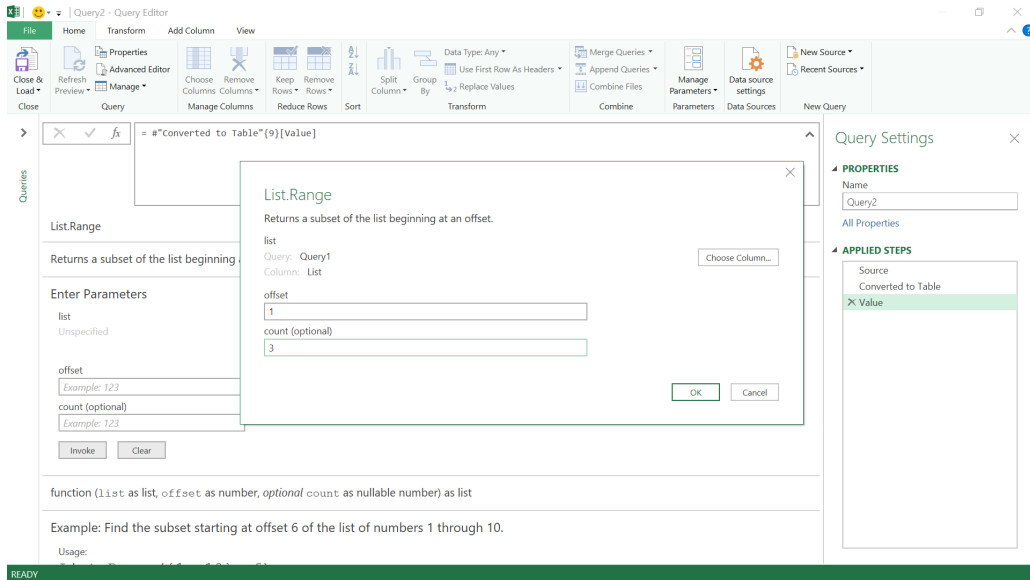
In the background, you can see a full description of the **List.Range** function, and we also have the option of trying out the function on one of my columns. Clicking the 'Choose Column' button accesses another screen:



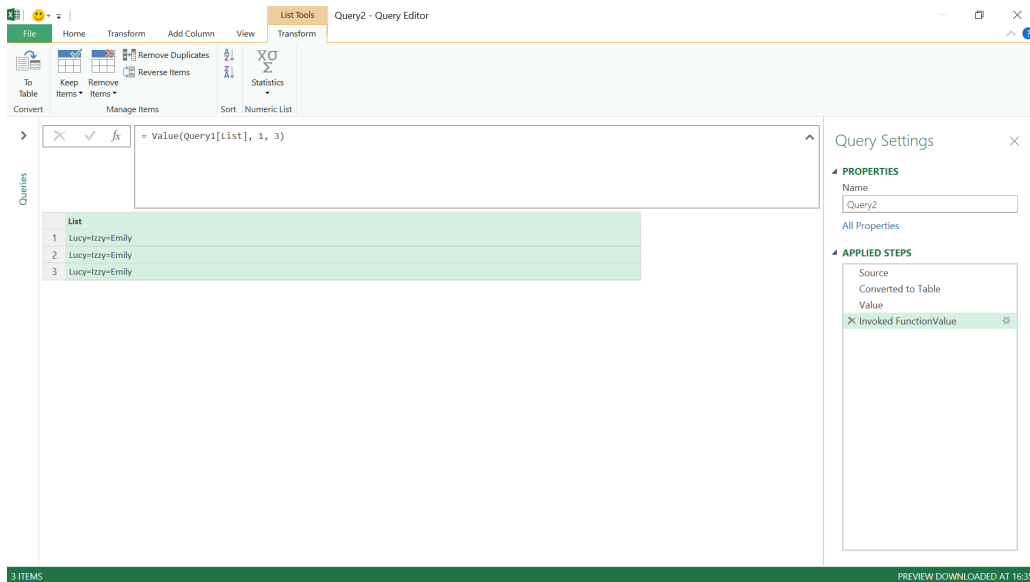
You may choose any query in my workbook to try out the function.



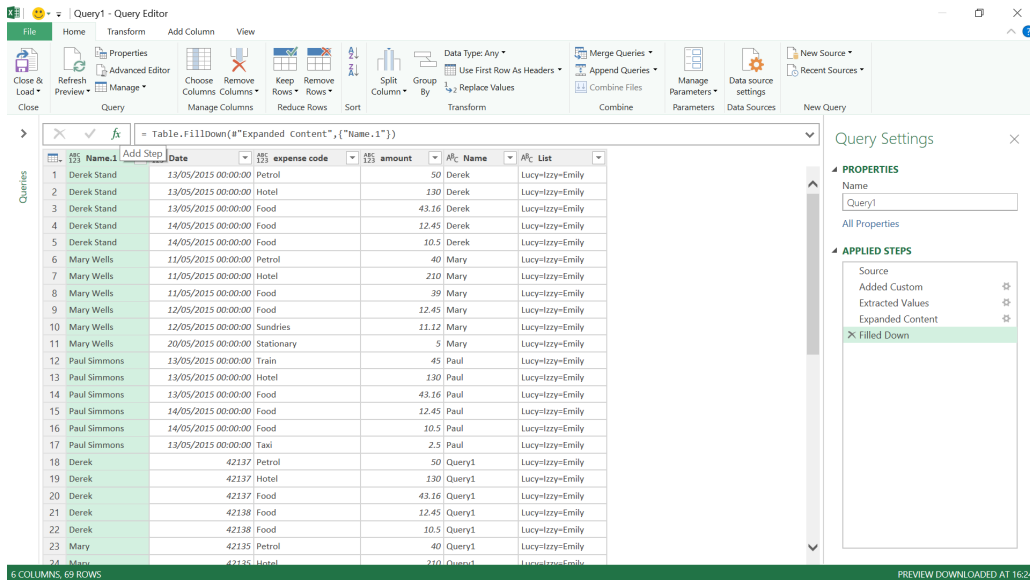
Once you pick a query, you can see all the columns in that query and the first few rows of data. Choose **List** as the column to use.



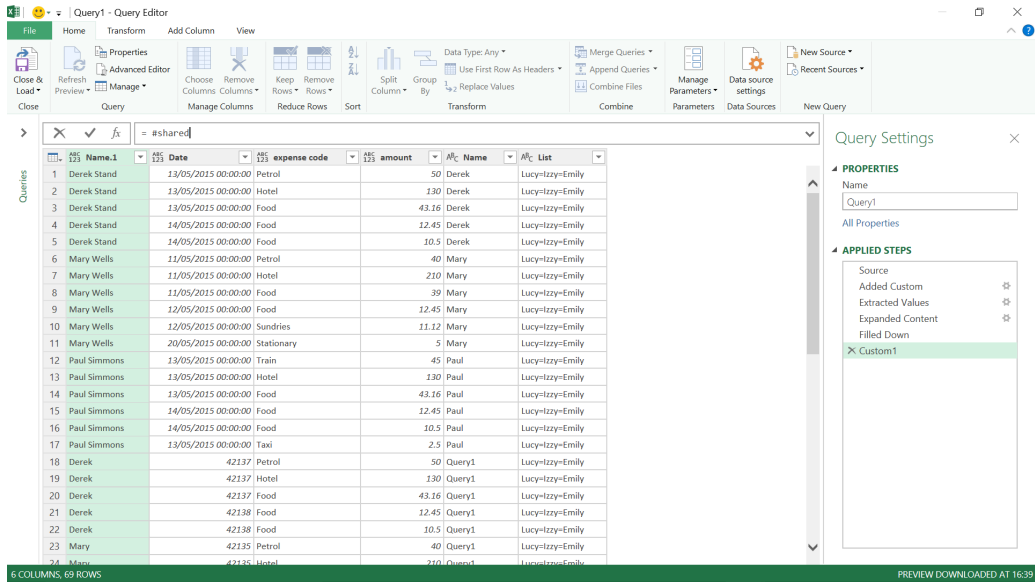
We are taken back to the previous pop-up screen so that I can enter the offset and count if I wish. Choose an offset of 1 and a count of 3 and continue.



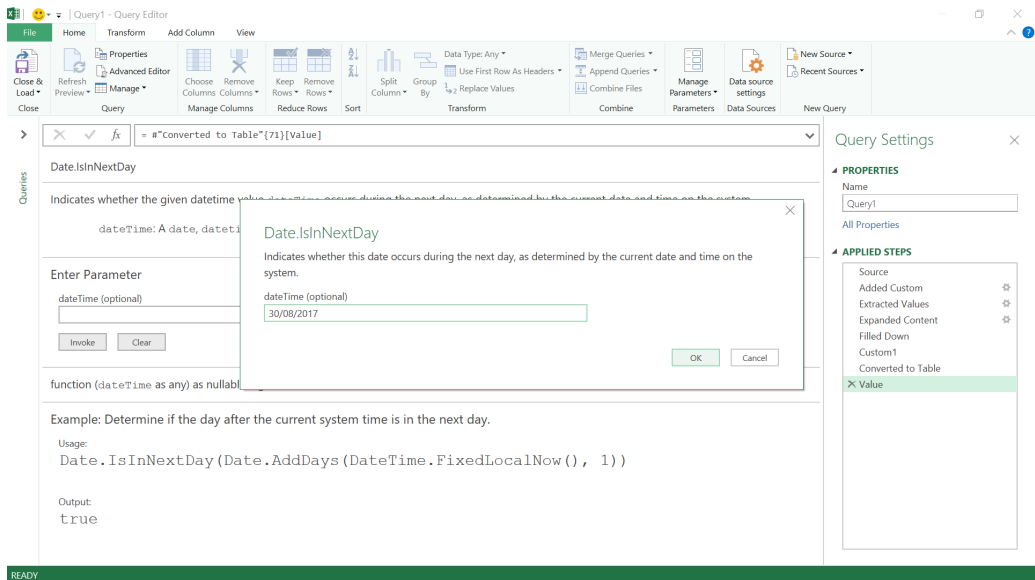
We have tested the **List.Range** function and it has come back with the first three entries in my list column. We can also call the **#shared** function from within a query to test out a function (or look for one). Choose to add a step by clicking the **fx** button in the Ribbon above **Query1**, as shown below:



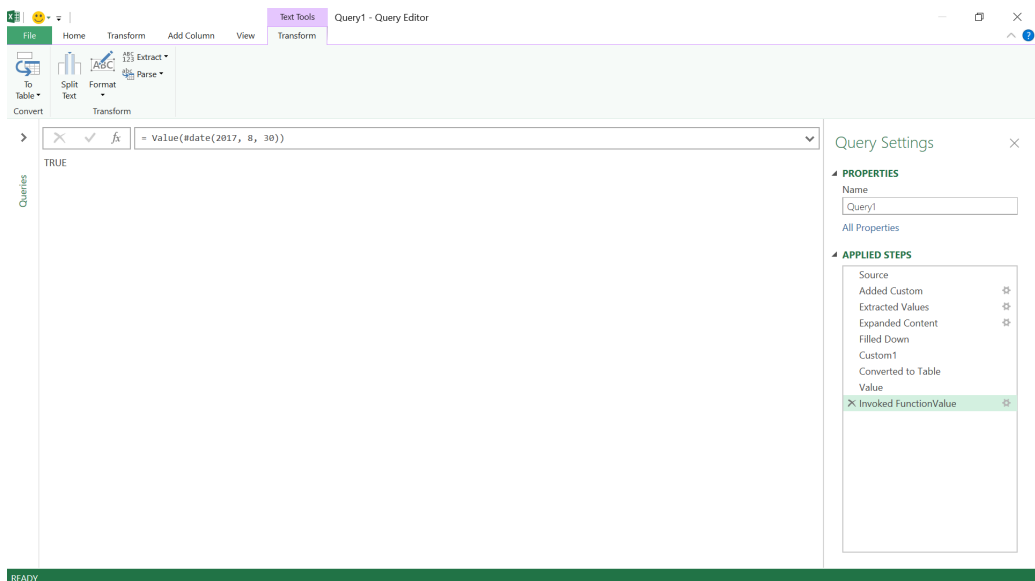
Now, choose to add the '#shared' step:



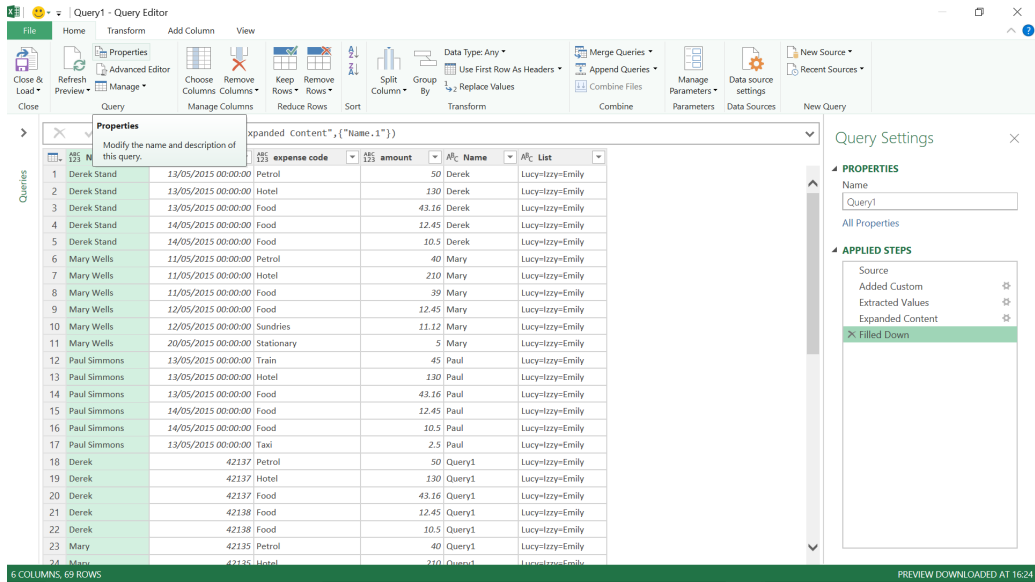
You can see the same record as before and can go on to try out a function against any data in the workbook. This time let's try something different:



Having acquired my answer, we may then continue with our query:



If we delete the extra steps we have created, from 'Custom1' to 'InvokedFunctionValue', you can get back to our original query.



More next month!

## Power BI Service and Mobile Latest Updates

Unfortunately, due to the earlier printing deadlines for the festive season, we have no Power BI Desktop updates for you this month. Try not to cry. No doubt we will be avalanched in the February newsletter!

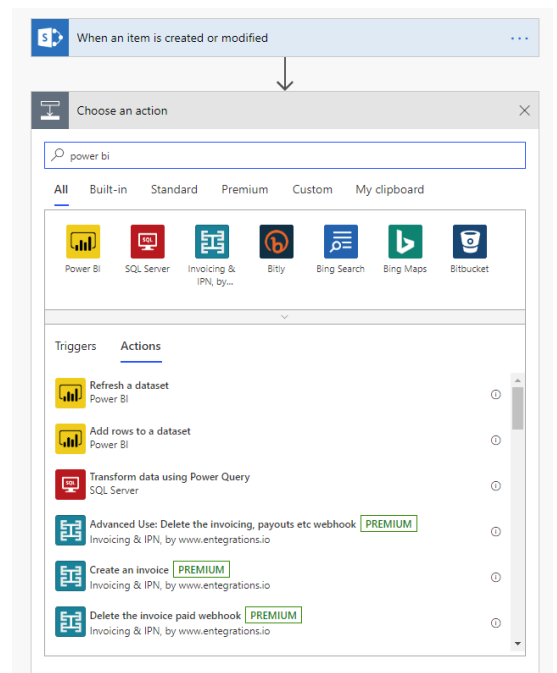
In the meantime, however, there have been quite a few updates for Power BI Service and Mobile. Here's the complete list of the latest updates:

- Refresh your Power BI dataset using Power Automate
- On-premises data gateway management via PowerShell
- Contact list for reports and dashboards
- Power BI lineage view (Preview)
- General Availability of Automated Machine Learning
- Large models in Power BI Premium (Preview)
- Data protection capabilities (Preview)
- Embed paginated reports in applications (Preview)
- Upgrade classic workspaces to new workspaces experience
- Update for On-premises data gateway
- Updates to Power BI mobile apps
- Roadmap updates.

Let's take a look at each in turn.

### Refresh your Power BI dataset using Power Automate

Microsoft has released a new Refresh dataset action for the Power BI connector for Power Automate (previously, Microsoft Flow). Now, you may trigger refreshes for your datasets in Power BI based on hundreds of Automate triggers. Whether your trigger is based on changes to items in your SharePoint list or updates to an Excel file in OneDrive or SharePoint Online or some complex day and time schedule, there are many instances where you can leverage this action.



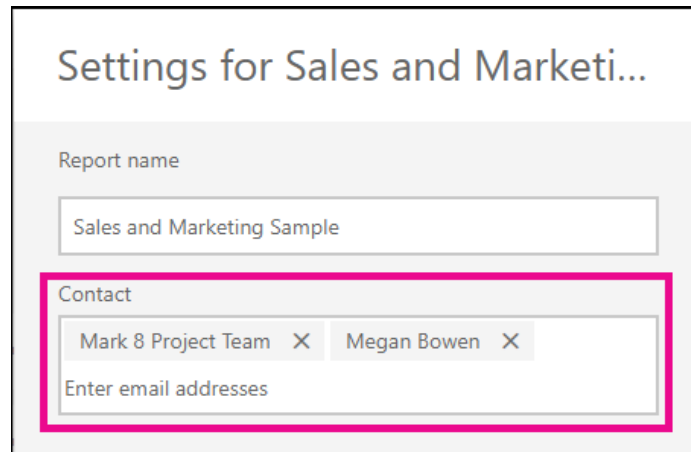
## On-premises data gateway management via PowerShell

This update has rolled a public Preview of PowerShell **cmdlets** that assist you in automating and managing tasks for Power BI data sources and On-premises gateways. These cmdlets are available in the PowerShell Gallery and can be installed in an elevated PowerShell session.

## Contact list for reports and dashboards

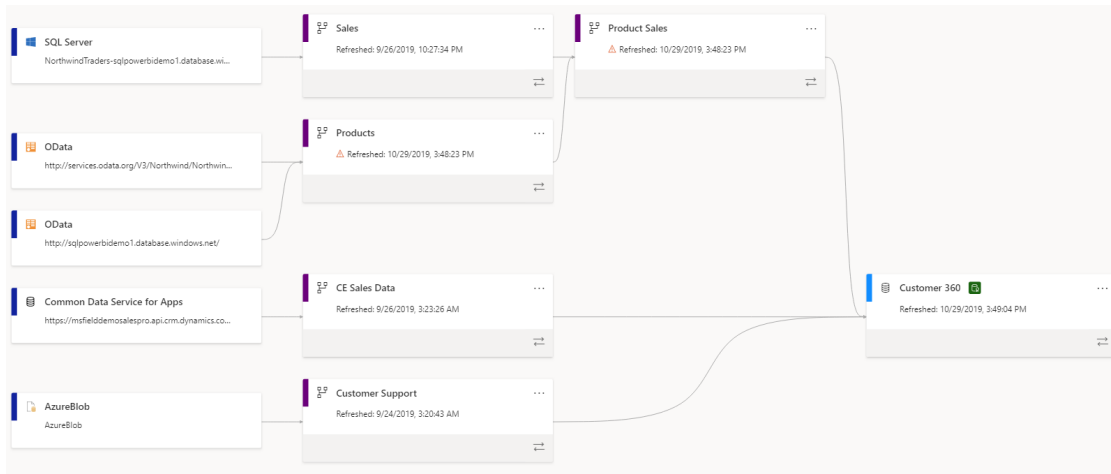
To help end users ask questions, raise issues with the data they see, or seek confirmation, Microsoft has added a new feature that allows authors to configure the list of contacts, which can include multiple

users and groups, for reports and dashboards in the Power BI Service. When used with the New Look for the Power BI Service, end users can quickly find and contact someone to make timely data driven decisions.



## Power BI lineage view (Preview)

Understanding the flow of data from the data source to its destination is critical for building reports. To help you visualise this, Power BI has introduced a public preview of lineage view. The view encompasses the flow of all Power BI workspace artefacts including dataflows, datasets, reports and dashboards, and their connections to the external data sources.



## General Availability of Automated Machine Learning

Microsoft has also announced that Automated Machine Learning (Auto ML) in Power BI is now Generally Available in all public cloud regions. With this feature, business analysts without a strong background in machine learning can build ML models to solve business problems

that once required data scientists. Most of the data science behind the creation of the ML models is automated by Power BI, while giving visibility into the process used to create your ML model to provide you with full insight.

## Large models in Power BI Premium (Preview)

Power BI Premium has had an upgrade with the public Preview of large models. Power BI datasets can store data in a highly compressed, in-memory cache for quick query performance. This enables fast user interactivity over large datasets. However, these dataset caches were limited to 10 GB after compression. The new large models update

remove this limitation, so dataset cache sizes are limited only by the Power BI Premium capacity size. You may check out <https://aka.ms/LargeModels> for further information such as availability by region, and how to set up your Power BI Premium capacity for large models.

## Large models in Power BI Premium

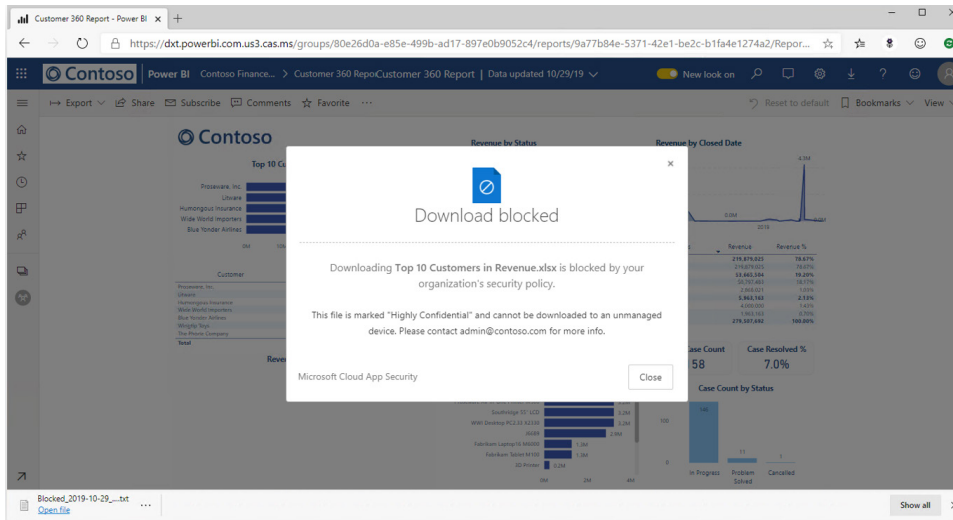


\* Up to available memory on your capacity, excluding memory required for query execution and processing, which is often 20-30% of total memory.

### Data protection capabilities (Preview)

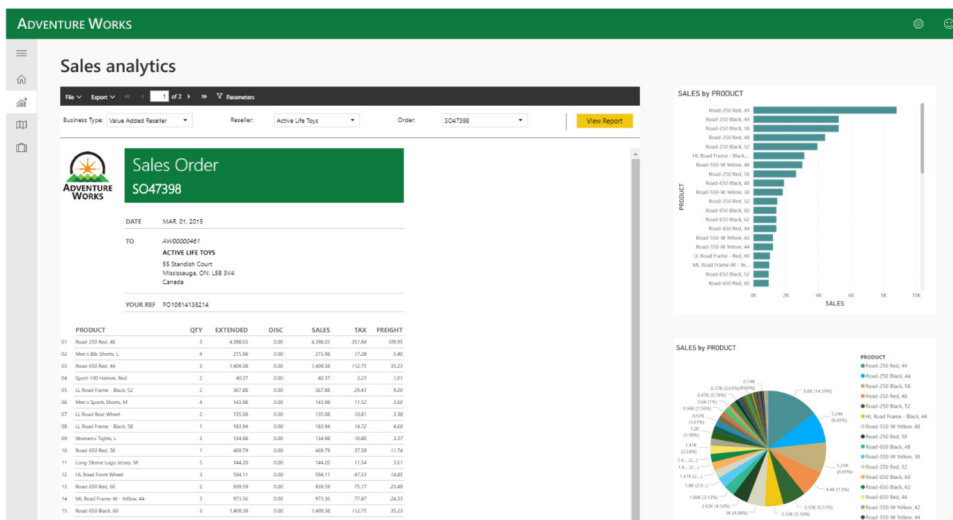
As enterprises grow their Power BI deployments, reducing the risk for data leakage becomes more and more critical. To help organisations govern their sensitive business data, the Power BI team has partnered with the Microsoft Information Protection and Cloud App Security team to release a public Preview of data protection capabilities for Power BI.

With this feature, you can now classify and label sensitive Power BI data using the familiar Microsoft Information Protection sensitivity labels used in Office, enforce governance policies even when Power BI content is exported to Excel, PowerPoint or PDF, and much more. These are also applicable to Power BI mobile apps.



### Embed paginated reports in applications (Preview)

This update also sees the public Preview of embedding paginated reports to allow independent software vendors (ISVs) and developers to embed pixel-perfect reports in their own applications using Power BI APIs.





## Upgrade classic workspaces to new workspaces experience

You may also now opt-in to upgrading “classic” workspaces to the new workspaces experience in Power BI. This Power BI new workspace experience has been General Available since April 2019. It offers improved security management, more flexible workspace roles and

new capabilities like a contact list to help you support your teams. Using the new workspace experience also enables several major features like shared and certified datasets and large models, to name a couple.

## Update for On-premises data gateway

These updates also improve the Power BI On-premises data gateway. Version 3000.14.39 includes a new way to change the gateway’s recovery password and an updated engine to match the Power BI Desktop November update.

## Updates to Power BI mobile apps

The Power BI mobile apps have also been updated with the following features:

- **Android support for remote configuration of Report Server access settings:** IT administrators can now use their organisation’s mobile device management (MDM) tool to remotely configure those settings on both iOS and Android devices
- **Cross-report drill through:** you may now tap on a data point to drill through to access other reports and report pages. When you drill through to a target page, the content in that page is filtered based on the drill through settings.

## Roadmap updates

Besides the updates listed above, Microsoft also takes the time to disclose upcoming features, not yet available. This month, they disclosed the following:

- **Decomposition tree:** this will be Generally Available soon, but no date advised as yet
- **Data lineage capabilities:** again, this will be Generally Available soon, but as above, no date specified yet
- **XMLA read / write:** a public Preview should be made available next month (February).

More soon, we’re sure!

## The A to Z of Excel Functions: DOLLAR

This function converts a **number** to text format and applies a currency symbol. The name of the function (and the symbol that it applies) depends upon your language settings. For example, with our Australian settings, this function converts a **number** to text using currency format,

with the decimals rounded to the specified place. The format used is **\$#,##0.00\_);(\$#,##0.00)**. For more information on custom number formatting, please see the link [here](#).



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

**DOLLAR(number, [decimals])**

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

- **number:** this is required and represents a **number**, a reference to a cell containing a **number** or a formula that evaluates to a **number**
- **decimals:** this argument is optional. This denotes the number of digits to the right of the decimal point. If **decimals** is a negative value, **number** is rounded to the left of the decimal point. If you omit **decimals**, it is assumed to be two (2).

It should be further noted that the difference between formatting a cell with a Ribbon command and using the **DOLLAR** function is that **DOLLAR** converts its result to text. A number formatted with the ‘Format Cells’ dialog box (**CTRL + 1**) is still a number. Microsoft claims that you can

continue to use the results generated by **DOLLAR** in other formulae, because Excel converts numbers entered as text to numbers when it calculates. This is not true though, as the following [UK] example demonstrates:

	A	B	C	D	E
1					
2		1		1	1
3		2		2	2
4		3		£3.00	=DOLLAR(3)
5		4		4	4
6		5		5	5
7	SUM	<u>15</u>		<u>12</u>	=SUM(D2:D6)
8					

Please see our example below:

	A	B	C
1	Data		
2	1234.568		
3	-1234.568		
4	-0.123		
5	1234.568		
6			
7			
8	Formula	Description	Result
9	=DOLLAR(A2,2)	Displays the first number in currency format, two digits to the right of the decimal point	£1,234.57
10	=DOLLAR(A2,-2)	Displays the first number in currency format, two digits to the left of the decimal point	£1,200
11	=DOLLAR(A3,2)	Displays the second number in currency format, two digits to the right of the decimal point	-£1,234.57
12	=DOLLAR(A4,4)	Displays the third number in a currency format, four digits to the right of the decimal point	-£0.1230
13	=DOLLAR(A5)	Displays the fourth number in a currency format, two digits to the right of the decimal point (default).	£1,234.57
14			
15		<i>Numbers displayed in £ sterling to highlight that DOLLAR is nothing to do with \$.</i>	

## The A to Z of Excel Functions: DOLLARDE

Whenever I see the **DOLLARDE** function, I think it should be called **DEUTSCHEMARK** (I like to leave my mark with these jokes...).



This function converts a dollar price expressed as an integer part and a fraction part, such as 1.02, into a dollar price expressed as a decimal number. These **fractional dollar** numbers are sometimes used for security prices.

The **fraction** part of the value is divided by an integer that you specify. For example, if you want your price to be expressed to a precision of 1/16 of a dollar, you divide the **fraction** part by 16. In this case, 1.02 represents \$1.125 ( $\$1 + 2/16 = \$1.125$ ).

This is a very different function to **DOLLAR**, which some users confuse with this function.

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

**DOLLARDE(fractional\_dollar, fraction)**

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

- **fractional\_dollar**: this is required and represents a number expressed as an integer part and a **fraction** part, separated by a decimal symbol
- **fraction**: this is also required. This is the integer to use in the denominator of the fraction.

It should be further noted that:

- if **fraction** is not an integer, it is truncated

- if **fraction** is less than 0, **DOLLARDE** returns the **#NUM!** error value
- if **fraction** is greater than or equal to 0 and less than 1, **DOLLARDE** returns the **#DIV/0!** error value.

Please see the example below:

	A	B	C
1	fractional dollar	fraction	
2	1.02	16	
3	1.1	32	
4	1.1	7	
5			
6			
7	Formula	Description	Result
8	=DOLLARDE(A2,B2)	Converts 1.02, read as 1 and 2/16, to a decimal number (1.125). Since the fraction value is 16, the price has a precision of 1/16 of a dollar	1.125
9	=DOLLARDE(A3,B3)	Converts 1.1, read as 1 and 10/32, to a decimal number (1.3125). Since the fraction value is 32, the price has a precision of 1/32 of a dollar	1.313
10	=DOLLARDE(A4,B4)	Converts 1.1, read as 1 and 1/7, to a decimal number (1.142857...). Since the fraction value is 7, the price has a precision of 15 significant figures (Excel maximum).	1.1428571428571
11			

## The A to Z of Excel Functions: DOLLARFR

This function is essentially the opposite of the **DOLLARDE** function: it converts decimal numbers to fractional dollar numbers, such as securities prices.



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

**DOLLARFR(decimal\_dollar, fraction)**

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

- **decimal\_dollar**: this is required and represents a decimal number
- **fraction**: this is also required. This is the integer to use in the denominator of a fraction.

It should be further noted that:

- if **fraction** is not an integer, it is truncated
- if **fraction** is less than 0, **DOLLARFR** returns the **#NUM!** error value
- if **fraction** is 0, **DOLLARFR** returns the **#DIV/0!** error value.

Please see our final example for this month below:

	A	B	C
1	fractional dollar	fraction	
2	1.125	16	
3	1.3125	32	
4			
5			
6	Formula	Description	Result
7	=DOLLARFR(A2,B2)	Converts the decimal number 1.125 to a number read as 1 and 2/16	1.02
8	=DOLLARFR(A3,B3)	Converts the decimal number 1.3125 to a number read as 1 and 10/32	1.1
9			

More Excel Functions next month...

## Upcoming SumProduct Training Courses

Location	Course	Date	Duration
Melbourne	Excel Tips and Tricks	13 Jan 2020	1 Day
Melbourne	Financial Modelling	14 - 15 Jan 2020	2 Days
Sydney	Power Pivot, Power Query and Power BI	15 - 17 Jan 2020	3 Days
Sydney	Excel Tips and Tricks	27 Jan 2020	1 Day
Melbourne	Excel Tips and Tricks	13 Jan 2020	1 Day
Melbourne	Financial Modelling	14 - 15 Jan 2020	2 Days
Sydney	Power Pivot, Power Query and Power BI	15 - 17 Jan 2020	3 Days
Sydney	Excel Tips and Tricks	27 Jan 2020	1 Day
Sydney	Financial Modelling	28 - 29 Jan 2020	2 Days
Sydney	Power Pivot, Power Query and Power BI	17 - 19 Feb 2020	3 Days
Melbourne	Power Pivot, Power Query and Power BI	2 - 4 Mar 2020	3 Days
Sydney	Excel Tips and Tricks	2 Mar 2020	1 Day
Sydney	Financial Modelling	3 - 4 Mar 2020	2 Days

## Key Strokes

Each newsletter, we'd like to introduce you to useful keystrokes you may or may not be aware of. This month, we thought we would consider **CTRL** and **ALT** with the function keys:

Keystroke	What it does
CTRL + ALT + F1	New macro sheet
CTRL + ALT + F2	Open
CTRL + ALT + F3	New name
CTRL + ALT + F4	Close application
CTRL + ALT + F5	Refresh all
CTRL + ALT + F9	Full recalculation
CTRL + ALT + F12	Thai dictionary

There are over 540 keyboard shortcuts in Excel. For a comprehensive list, please download our Excel file at [www.sumproduct.com/thought/keyboard-shortcuts](http://www.sumproduct.com/thought/keyboard-shortcuts). Also, check out our new daily **Excel Tip of the Day** feature on the [www.sumproduct.com](http://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](mailto: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](http://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](mailto: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](mailto:training@sumproduct.com) for a copy of the brochure or download it directly from <http://www.sumproduct.com/training>.

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

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