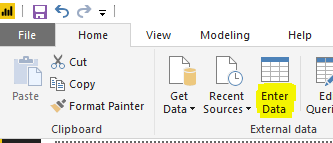
Of all the incredible features in Power BI, my favorite one (which ironically I do not get to use very often) is the combination of the DAX expressions SELECTEDVALUE and SWITCH.

If I ever need slicer selections that are not quite so readily available in a column of my dataset, this combination is perfect. Personally, the end users at my company absolutely love to use slicers. Considering the similar functionality to Excel filters, which seem very commonly used even for those users not as tech savvy, this DAX feature is so immensely powerful and useful for one distinct reason: Anyone can use it.

In short, the elevator speech for this DAX combination is that it allows you to customize slicers in any way imaginable across multiple looks or datasets. In turn, you can create even more powerful and all-encompassing views for your end users, all with the use of a generic slicer.

Here is how it works:

**Step 1: Enter Data**



This is where you enter the slicer values that you want. Later on, the selection will retrieve all values within the context of your slicer selection here, all of which you customize behind the scenes.

In this example, I will use dummy financial data that Microsoft has provided specifically for Power BI practice, available [here](https://docs.microsoft.com/en-us/power-bi/sample-financial-download). To keep it simple, I will use the Country column for this example. The dataset consists of the following countries:

* United States of America
* Canada
* France
* Germany
* Mexico

Now, after pressing Enter Data, I will put in the values United States and International (assuming the company is based in the United States), shown below:



**Step 2: Create Measures**

For each value I entered in step 1, I will create a custom measure, eventually getting used in the SELECTEDVALUE measure. Let’s say we wanted to calculate total profit of domestic vs. international sales.

Again, this is where the value of this combination comes into play; I can tailor customizable text for all slicer selections for my end users of this report, and simultaneously I can tailor the DAX measure used to pull the data for each individual selection.

This example is rather straightforward, but here is what my measures are:

Domestic = CALCULATE(SUM('financials'[Profit]),'financials'[Country]="United States of America")

International = CALCULATE(SUM('financials'[Profit]),'financials'[Country]<>"United States of America")

**Step 3: Create SELECTEDVALUE Function**

Now that we have everything set up, we create the final function. We will also utilize DAX variables, which can be incredibly useful in organizing DAX formulas efficiently. Here is the end result, which I will break down piece by piece:

SlicerSelection =

VAR Selection =

SELECTEDVALUE ( 'SelectedValue'[Selections] )

RETURN

SWITCH (

TRUE (),

Selection = "Domestic", [Domestic],

Selection = "International", [International],

CALCULATE ( SUM ( 'financials'[Profit] ) )

)

* **VAR Selection =  
   SELECTEDVALUE ( ‘SelectedValue’[Selections] )  
  RETURN**

This is where I set the variable I titled ‘Selection’. Instead of having to re-type the SELECTEDVALUE formula multiple times (especially when you may have well more than two selections), this makes it much easier to look at, and much less likely to have a typo in the formula. Now, I can simply type ‘Selection’ and the DAX measure will automatically reference the SELECTEDVALUE formula every time.

Whenever you add variables into a formula, you must always add RETURN after you add variables so you can signify where the actual DAX function will begin.

You also have the option after ‘SelectedValue’[Selections] to add an alternate result. This is optional, but can be very useful if you ever utilize this where there may be more than one value. By default, this value is BLANK().

* **SWITCH (  
  TRUE(),**

For anyone proficient in Excel, this is essentially the same as nesting a number of IF statements all at once. This combination of SWITCH(TRUE() will simply look for a value that is TRUE, which in this case will be the slicer selections.

**\*Note: Without using SELECTEDVALUE, you can also use this combination of SWITCH(TRUE() to provide very customizable values on its own, including text values even as a measure. But if any options are TRUE in multiple instances, the order will matter as the measure will provide the first TRUE value available.\***

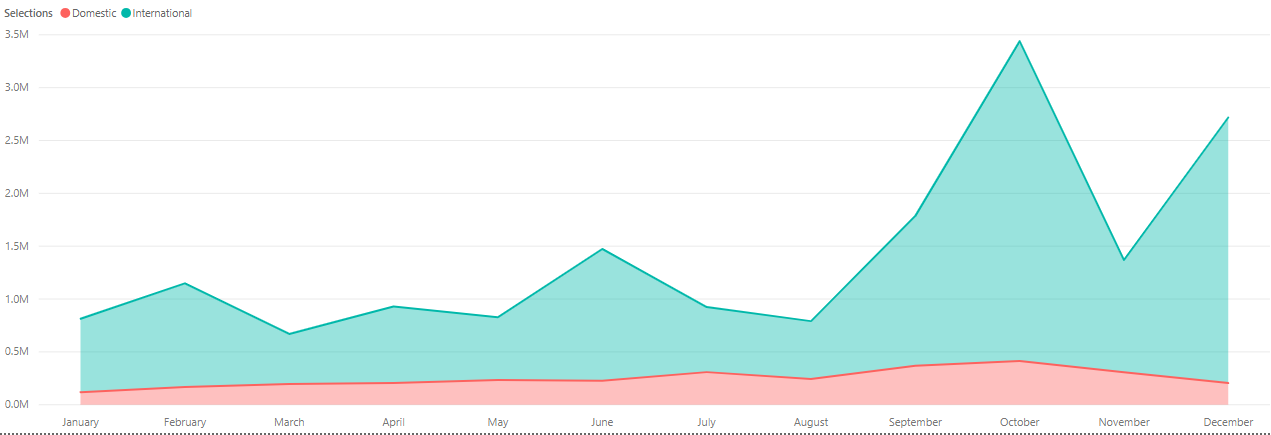
* **Selection = “Domestic”, [Domestic],  
  Selection = “International”, [International],  
  CALCULATE ( SUM ( ‘financials’[Profit] ) ) )**

This is where we combine steps 1 and 2 together.

If our variable ‘Selection’ returns the text value of “Domestic” (or if the value is TRUE), then return the measure we created for [Domestic]. Same goes for International. For the final line, that is our calculation if none of the ‘Selection’ values return TRUE (or if no slicer selections were chosen). This means that if the slicer has nothing selected, it will calculate the Profit column like a normal measure.

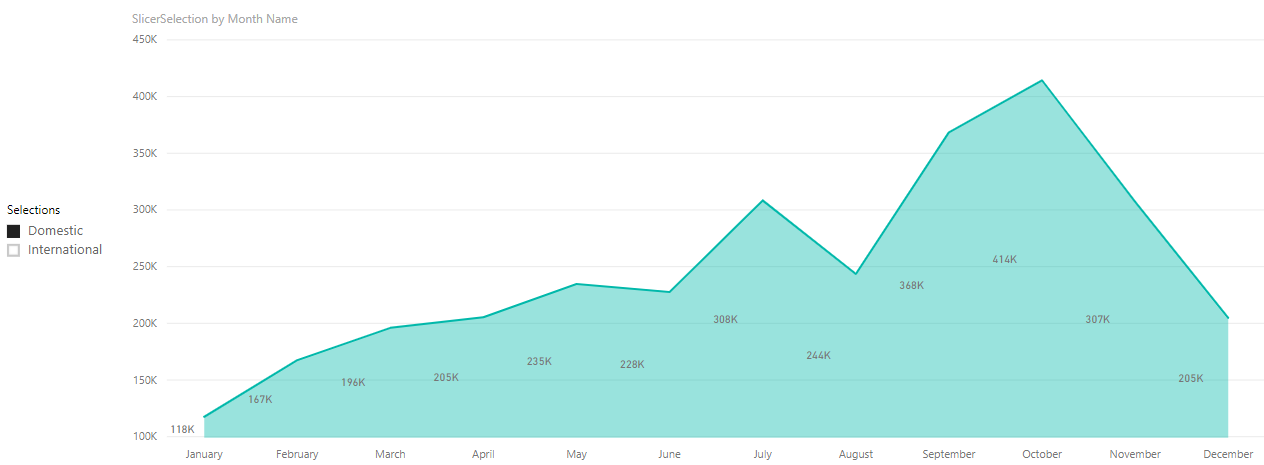
Since I cannot state this enough, every single aspect of this functionality is customizable so far.

Within seconds, I can now add a stacked area chart for example, showing my domestic profit and my international profit by month:

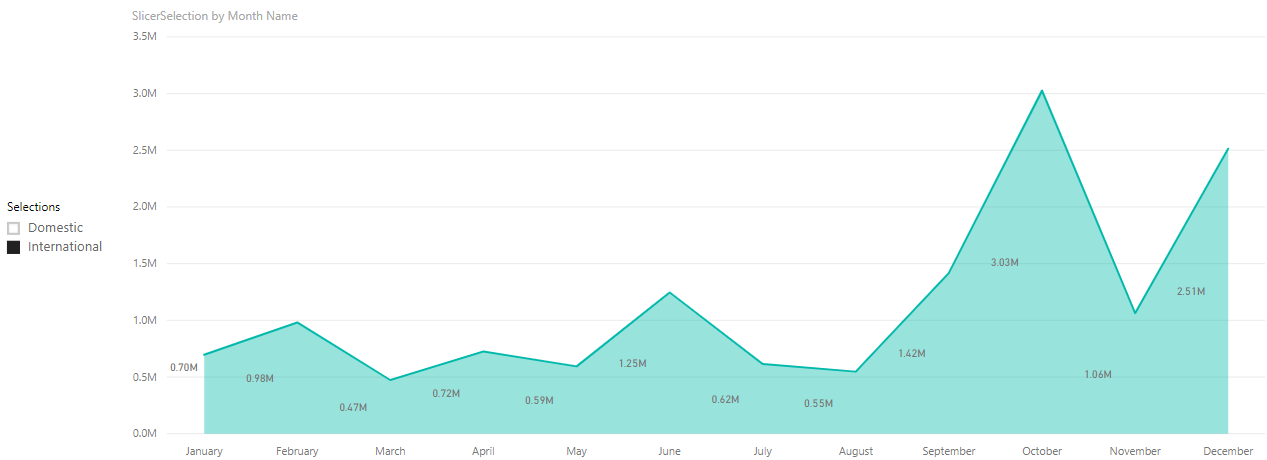


Or, I can simply create a slicer with our values from step 1, shown below:

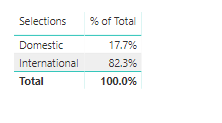
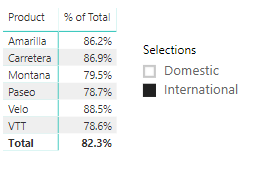
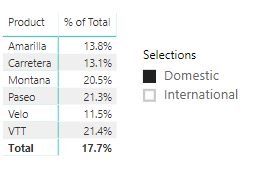
Domestic:



International:



Or, even just a simple table of the % of total profit that is attributed to Domestic vs. International sales, based off of our fancy new slicer:



This is an extremely watered down version of what this combination of DAX functions is capable of. If you have a report with multiple datasets at once, you can create a single slicer like this with highly customized measures pulling values across these datasets, rather than a slicer based off one specific column within the dataset.

No matter what your end users are looking for, and more importantly no matter how technically proficient they may be, the combination of SELECTEDVALUE and SWITCH can provide an enormous amount of value to them in very little time and processing power. Even with something as simple as a slicer.