5 *Thumb Rules for* DAX Measures

- **No Implicit Measures** – Meaning, dragging a field (column) into values area of the pivot to get Sum, Count etc... *(that's illegal)*. You shouldn't do that in DAX, instead for every SUM, COUNT or AVERAGE to be calculated, write a DAX formula i.e. create an Explicit Measure. Implicit measures won't take you far and won't perform sophisticated calculations
- **No Calculated Columns** – Whenever in doubt, decide NOT to make a calculated column instead create a Measure. Calculated columns are legal when you want the values on rows, columns, slicer or horizontal axis of the chart etc.. Calculated Columns add major bulk to your data model
- **No Naked Columns** in a measure – Writing the following DAX =Data[Units] makes no sense *(and will result in an error)*, because it doesn't know what to do with the entire Units column. Always* use an aggregator around the column. This will work =SUM(Data[Units])
- **Measure Writing Conventions** – While writing DAX it's a best practice to
  - Precede column name with table name in square brackets. Data[Category]
  - Measures to be written without table name in square brackets. [Total Sales]
- **Generating a Scalar Value** – This is super important and you'll often run into errors because your measure isn't returning a scalar value in other words a single value. Your measures have to return a single value else they'll throw up an error

*There can be a possibility of writing naked columns in row context in a row iterator function like SUMX. For instance =SUMX(Data, Data[Units] * 5) will multiply each row's unit to 5 and will then take the SUM. Two worthy things to notice. 1. Data[Units] column is naked and is working under row context 2. There is no point of this calculation, it's just a demo. I couldn't come up something meaningful promptly ☹

- Always use a separate Date table in your data model. Mark it as a Date Table
- Hide columns that are needed but are irrelevant for the user. Quick Tip: Use Relationships view to hide multiple columns: use SHIFT + DOWN ARROW for selection of columns >> Right click menu >> Hide
- Create a Blank Table and assign all Measures to that Table. This way your measures won't be scattered all over the model
- Use the new model diagram to break-out complex models by subject area in separate diagrams
- Use DIVIDE() function to prevent division by 0, and to improve the speed of your divisions
- Use standard abbreviations like YTD, LY, PY, PP as suffix, to keep the base fields together
- Use double backslash // to write comments in a row, in between your DAX code
- Some important operators and their explanation

Operator	Meaning	Example
&& (double ampersand)	Creates an AND condition between two expressions	Cal [Year] = 2019 **&&** Cal [Month] ="Jan"
|| (double pipe symbol)	Creates an OR condition between two expressions	Sales [Region]= "North") **||** Product [Colour] ="Red"
IN	Creates multiple OR conditions	Product [Category] **IN** { "Low", "Mid", "Premium" }

HOW DAX MEASURES ARE EVALUATED

Between the DAX Code and it's result is a Black Box. You need to understand these 3 steps thoroughly to get your head around what goes inside that box

Consider this Data and Pivot

Tran Id	Date	Region	Sales	Channel
101J11	02-Jan-11	New Delhi	45	Promotional
102J11	02-Jan-11	Mumbai	0	Promotional
103J11	03-Jan-11	New Delhi	174	Promotional
104J11	03-Jan-11	Mumbai	18	Affiliate
105J11	04-Jan-11	Mumbai	0	Affiliate
106J11	06-Jan-11	Mumbai	0	Promotional
107J11	06-Jan-11	New Delhi	60	Organic
108J11	06-Jan-11	Mumbai	89.91	Affiliate
109J11	07-Jan-11	New Delhi	24	Promotional
110J11	08-Jan-11	New Delhi	83.88	Affiliate

Region	Total Sales
Bangalore	5308
Mumbai	20969
New Delhi	22834
Grand Total	**49111**

Let's talk about how the total sales for Bangalore is calculated

Step 1 : The filter context from the pivot / visual plus the CALCULATE function filter context is applied to the data. For Total Sales (5,308) the filter context is Region = Bangalore

Tran Id	Date	Region	Sales	Channel
124J11	25-Jan-11	Bangalore	45	Promotional
126J11	28-Jan-11	Bangalore	22.17	Promotional
137F11	15-Feb-11	Bangalore	41.94	Promotional
149M11	01-Mar-11	Bangalore	30	Promotional
153M11	09-Mar-11	Bangalore	0	Affiliate
157M11	12-Mar-11	Bangalore	54	Affiliate
175A11	04-Apr-11	Bangalore	27	Affiliate
216M11	25-May-11	Bangalore	15	Promotional
225M11	30-May-11	Bangalore	0	Affiliate
231J11	11-Jun-11	Bangalore	6	Affiliate

Region is filtered to Bangalore.

Note that filter context can also come from the lookup table and or the CALCULATE function

Step 2 : On the filtered data, the DAX calculation is carried out. In our case the DAX Measure for Total Sales = SUM (Data[Sales])

Step 3 : The result of the DAX calculation is then returned to the visual / pivot. In our case the result for Bangalore Sales is 5,308. This process then repeats for all the coordinates in a pivot table / visual. In our case there are 4 coordinates, Region = Bangalore, Region = Mumbai, Region = New Delhi, Region = All (no filters) which is also the grand total

Words of Wisdom

- The above 3 steps happen under the hood and what you see is just the result. To be able to write, edit and debug complex DAX measures you'll have to master and mentally visualize how these 3 steps are working on your data
- I have said it earlier.. I'll say it again – your measure has to compulsorily return a scalar value
- For mastering DAX focus on understanding behavior of functions - What does it do, What does it accept, What does it return and if there are any exceptions. Rest will fall in place with practice

RELATIONSHIP TIPS *in power bi (mind it ☺)*

What are Relationships

- Relationships are like virtual VLOOKUPs – A VLOOKUP without actually performing it
- Create relationship between 2 tables – In the Relationship Tab of Power Pivot or Power BI simply drag the common column from the transactions table *(the data-set where you thought of writing a VLOOKUP)* and link it to the matching column in lookup table *(source data for VLOOKUP)*

Relationship Pitfalls

1. Always Always Always create the relationship from the transactions table to the lookup table and NOT the opposite way

2. Most times Power BI will create the relationship automatically, always recheck them manually

3. At times you'll have the need to relate one column *(of the lookup table)* to multiple columns *(of the transactions table)* Sure enough… you can do that but there can only be one active relationship between 2 tables. The second relationship that you create will be inactive *(with a dotted line)*. The dotted line marks the inactive relationship

4. The inactive relationship can be made active in measures using USERELATIONSHIP function. USERRELATIONSHIP (<column1>,<column2>), this can be used in the second argument of the CALCULATE function

5. Duplicates are not allowed in the lookup table (both in Power Pivot and in Power BI). You'll get an error, if while creating the relationship there are duplicates in the lookup table

6. If duplicates pop up after refreshing the data model – it would still result in an error and all measures and filter propagation would stop working

7. The following types of relationships are supported in Power BI/ Power Pivot

Relationship	Power Pivot	Power BI	Notes
One to Many	Yes	Yes	One of the most common type and is supported in both
One to One	No	Yes	Power Pivot automatically converts it into one to many
Many to Many	No	No	There are hacks to make it happen but there is no default option