DAX

Agenda

- ➢ Introduction DAX
- ❑ DAX & Power BI
- ❑ DAX Functions
- ❑ DAX Variables
- ❑ DAX Best Practice

What is DAX

- DAX – **D**ata **A**nalysis E**x**pression – a formula language defined by
  - Functions
  - Operators
  - Constants
- Commonly associated to Excel Power Pivot, **Power BI**, SQL Server Analysis Services (SSAS) Tabular Model
- Used to build custom calculated columns and fields
- Enhances your data model
- It's a broad language - is **not** a programming language – it's a formula language
- Has **Two primary** data types – **Numeric** (integers, decimal & currency) and **Other** (string, binary objects)
- **Operator Overload** – we can mix data types in a calculation; result will change based on inputs

Goal – give you the building blocks for DAX to allow you to explore

DAX - Functions

- DAX function – inbuilt functions provided in DAX language enabling users to perform various actions.
- Some functions are like Excel's functions, but the functionality is different.
- DAX functions are column or table driven while Excel functions are cell or range of cells driven
- Calculation modification maybe required to interchange between Excel functions and DAX functions.

DAX Operators

- Common MATH Operators
- Priority defines the sequence of execution

PRIORITY LEVEL	OPERATOR	DESCRIPTION
1	()	Parentheses – grouping
1	F()	Scalar functions
1	IN	Inclusive OR list
2	^	Exponentiation
3	+, –	Sign – unary plus/minus (-1)
4	*, /	Multiplication, division
5	NOT	Logical negation
6	+, –	Addition, subtraction
7	&	Text concatenation
8	=, ==, <>, <, >, <=, >=	Comparison operators
9	&&	Logical AND
10	||	Logical OR

armely

DAX Calculation

- DAX has two primary calculations
  - Calculated Column
  - Measures
  - Calculated Tables
  - ** Can also be used to defined Row-Level-Security (RLS) rules, - expression that enforces filters over model tables.

Calculated Columns

- DAX can be used to create calculated column
- Formula evaluated for each table row and a single value is returned
- When added to Import Storage Mode - evaluated when data model is refreshed.
- When added to DirectQuery Storage Mode- evaluated when data is queried.

  Example FullName = TableName[FirstName] &” “& TableName[LastName]

Measures

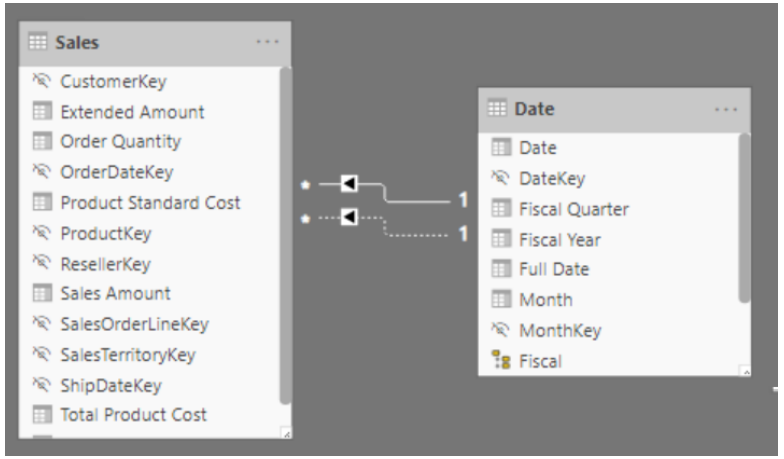
- DAX formula for measures can be on any table
- Formula goal is achieving summation over model data
- Returns single value
- Measures are evaluated on query time; results are never stored.

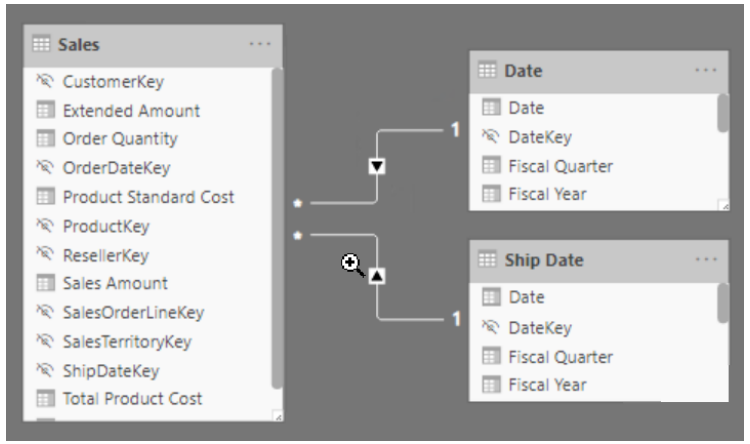
  Example Profit = SUM([Revenue]) – SUM([Cost])

Calculated Tables

- Using formula to create a series of data which produces a table.
- Examples –
  - **Date Table** – required to allow time filters [Time Intelligence]
    - If source doesn't include date table, you can create one using **CALENDAR** or **CALENDARAUTO** DAX functions

    CALENDAR(<start_date>, <end_date>)
  - **Role-playing Dimension** – when data model has multiple relationships (sales table with **order date** and **shipping date** both related to date)





armely

DAX Syntax

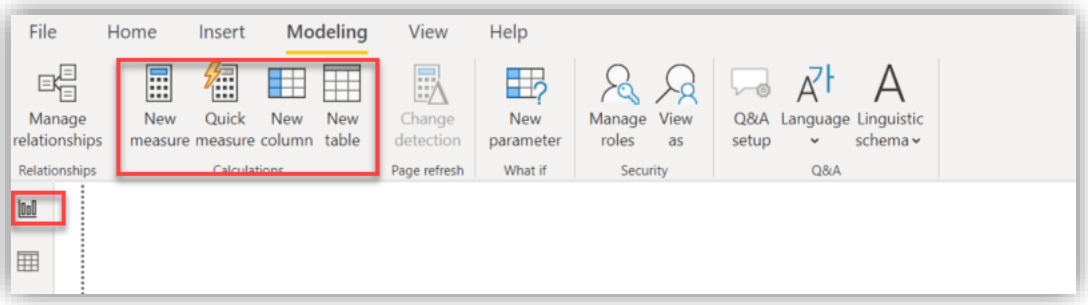
- DAX Formula always starts with **=** after that any expression can be provided.
- Its **case insensitive** SALES and Sales mean the same thing
- When using table or column as input to a function, it must be fully qualified **'Table Name'[column name]**
- Measures
  - Must always be in [] bracket
  - Must be unique within the model – you can't have [Total Sales] in two locations.
- Columns
  - Must be unique in the context table – multiple tables can have same column name

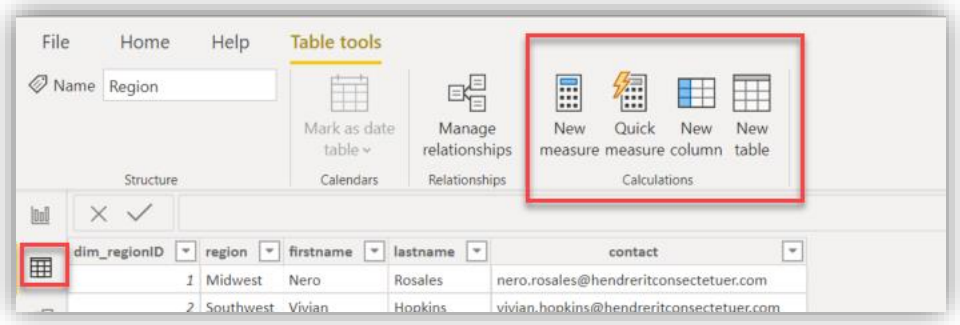
Agenda

- ✓ Introduction DAX
- ➢ DAX & Power BI
- ❑ DAX Functions
- ❑ DAX Variables
- ❑ DAX Best Practice

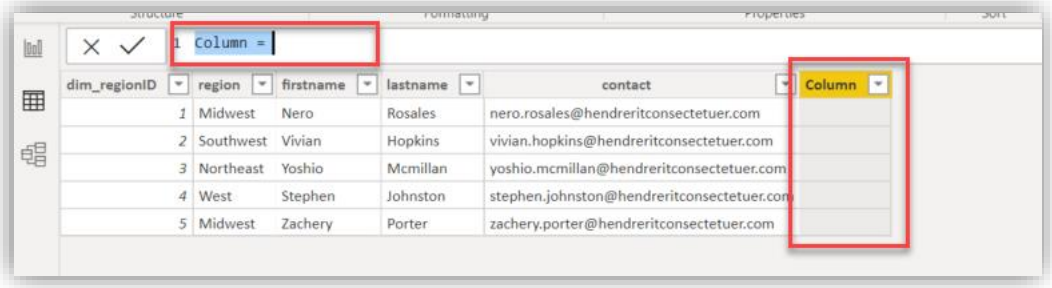
DAX & Power BI

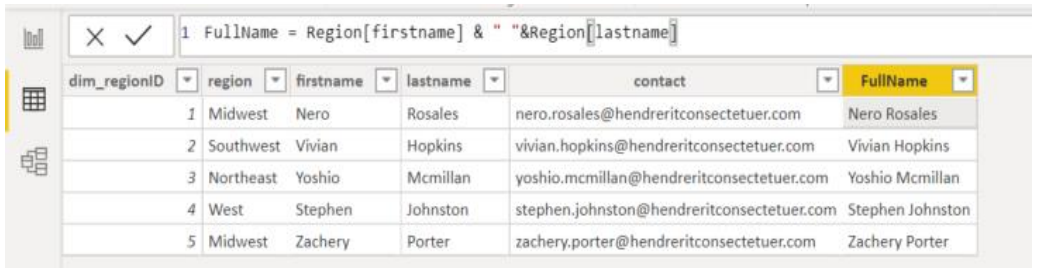
- Calculations accessible from **Modeling** in **Report** or **Table tools** in **Data**





- Adding a new column





DAX & Power BI

- Adding New Measures

```
%Profit = DIVIDE(sum(Sales[revenue]), sum(Sales[cost]))
```

dim_marketID	dim_customerID	date_id	dim_productcategoryID	dim_productsID	dim_regionID	units_sold	cost	revenue	Profit %	Diverge_Profit
2	62	37	2	6	1	270	$2,809	$6,344	225.85%	2.26
2	37	2	2	2	1	1997	$2,560	$5,289	206.60%	2.07
4	26	21	2	1	1	1670	$4,714	$7,689	163.11%	-1.63
4	7	29	2	3	1	553	$3,564	$9,690	271.80%	2.72

- Adding Calculated Table

```
Sample_Date_Table = CALENDAR("01/01/2020", "12/31/2020")
```

Date
1/1/2020 12:00:00 AM
1/2/2020 12:00:00 AM
1/19/2020 12:00:00 AM
1/20/2020 12:00:00 AM
1/21/2020 12:00:00 AM
1/22/2020 12:00:00 AM

Fields

Search

Sales

Sample_Date_Table

survey

Calculated Columns, Measures & Table DEMO

armely

Agenda

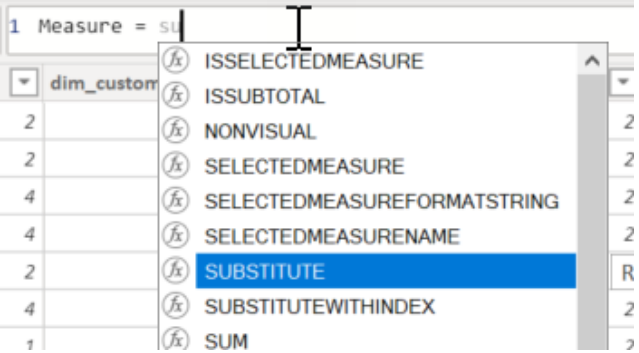
- ✓ Introduction DAX
- ✓ DAX & Power BI
- ➢ DAX Functions
- ❑ DAX Variables
- ❑ DAX Best Practice

DAX Functions… are many

Grouped as

- Aggregations
- Counting
- Logical
- Information
- Text
- Date

- Power BI offers intelliSense when typing the functions



armely

Aggregations

- SUM, Average, MAX, MIN etc.. - Operates over a **single column** and has no awareness of individual rows in the column (evaluates all rows in a column) example: Revenue = SUM(Sales[Revenue])

- SUMX, AVERAGEX and others with X function – these functions iterate through the table and evaluate the expression on **multiple columns** and can complete **row by row** evaluation.
  - Example Texas Revenue = SUMX(Filter(Sales, RELATED(Prodctcategory[name])= “AI”),Sales[Revenue])

armely

Counting Functions

- Count, DistinctCount, CountRows, CountBlanks
- CountA – counts cells that are not empty
- CountAX – nonblank while evaluating an expression
  - COUNTAX(FILTER(‘Customer’, [State]=“Texas”), [State])

Logical Functions

They return TRUE/False based on logic performed

AND, OR, NOT, IF, IFERROR – also expressed as operators (&&, ||, <>)

Example Syntax = IFERROR(4/0), “Div by 0”)

returns “Div by 0”

Information functions

- ISBLANK, ISNUMBER, ISTEXT, ISNONTEX, ISERROR – looks at the value or column provided and tells if the value matches expected type.
- Data type knowledge is important.

Text Function

- SEARCH, CONCATENATE, FORMAT, RIGHT, LEFT, LEN, REPLACE – Works with tables and columns evaluating a string value. Can also be used to control date format
  - Example Syntax = FORMAT(TODAY(), “MM/DD/YY”)

Date Functions

- DATE, HOUR, NOW, TODAY, EOMONTH, WEEKDAY – Evaluates date and time function

Example Syntax = EOMONTH(TODAY(), 0) - returns the last day of current month

Time Intelligence

- Time intelligence functions enables to analyze data using time periods, days, months, quarters years.
- DATEADD, DATESBETWEEN, DEATESINPERIOD, SAMEPERIODLASTYEAR, STARTOFMONTH, PARALLELPERIOD, TOTALYTD…

Example Syntax = CALCULATE SUM(SALES[REVENUE]), SAMEPERIODLASTYEAR([DATE]) – Returns revenue shifted one year back from dates in date column.

Functions DEMO

Agenda

- ✓ Introduction DAX
- ✓ DAX & Power BI
- ✓ DAX Functions
- ➢ DAX Variables
- ❑ DAX Best Practice

armely

DAX VARIABLES

- Very powerful and increases the readability and reusability of your code
- Varname = returnedvalue

Example

```
Syntax =
var sales_margin = SUM(SALES[REVENUE])-SUM(SALES[COST])
RETURN
IF ( Sales_margin > 1000, Sales_Margin*0.3, Sales_margin*0.2)
```

Agenda

- ✓ Introduction DAX
- ✓ DAX & Power BI
- ✓ DAX Functions
- ✓ DAX Variables
- ❑ DAX Best Practice

Best Practice

- No spaces in table names
- Always include the table name in formulas (don't omit it, even though DAX lets you)
- When reference a measure, avoid using table name
  - Use [Profit] instead of Sales[Profit]

Agenda

- ✓Introduction DAX
- ✓DAX & Power BI
- ✓DAX Functions
- ✓DAX Variables
- ✓DAX Best Practice