DAX Functions

Functions in DAX / DAX Functions Categories

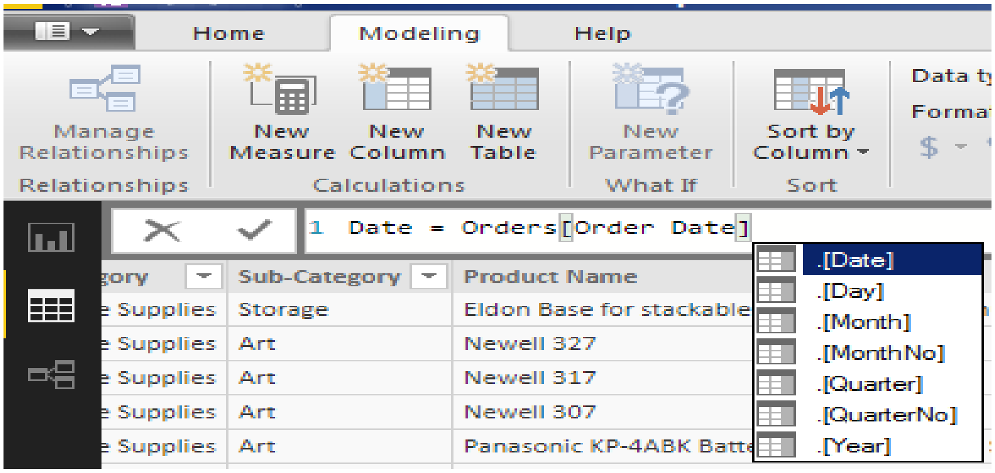
DAX provides the following types or Categories of functions which we mostly used.

- Date and Time Functions
- Logical Functions
- Text Functions
- Math & Statistical Functions
- Filter Functions
- Time Intelligence Functions

Basic Date & Time Functions

Mostly we used Date and Time functions to create Calculated Columns / New Columns. Let's us discuss about below date functions.

- YEAR
- MONTH
- DAY
- WEEKDAY
- WEEKNUM
- FORMAT (Text Function) → Month Name, Weekday Name
- DATE
- TODAY
- NOW
- HOUR
- MINUTE
- SECOND
- TIME
- DATEDIFF
- CALENDAR
- EOMONTH
- STARTOFMONTH



```
1  Date = Orders[Order Date]
```

gory	Sub-Category	Product Name
Supplies	Storage	Eldon Base for stackable
Supplies	Art	Newell 327
Supplies	Art	Newell 317
Supplies	Art	Newell 307
Supplies	Art	Panasonic KP-4ABK Batte

Creating Calculated Columns form Date Fields

Here we will create fields for Year, Month and Day.

a) Year = Year (Orders [Order Date])
b) Month = Month (Orders [Order Date])
c) Day = Day (Orders [Order Date])
d) Week Day = Weekday (Orders [Order Date])
e) WeekNum = WEEKNUM (Orders [Order Date])

Create the Month Name field

Month Name = Format (Orders [Order Date],"MMMM") → Gives full Month Name
Month Name = Format (Orders [Order Date],"MMM") → Gives Month Name abbreviations

Create a Weekday Name field

Weekday Name = FORMAT (Orders [Order Date],"DDDD") → Gives Full Weekday Name
Weekday Name = FORMAT (Orders [Order Date],"DDD") → Gives Weekday Name abbreviations

DATEDIFF

No of Days = DATEDIFF (Orders [Order Date], Orders [Ship Date], DAY)
(Select don't summarize for No of Days to get correct results)

Calculated Tables

In a calculated table, the table values are generated by Data Analysis Expression (DAX) and the values are stored in the Power BI model. Usually using Calculated Tables, we will create Date Dimension table and use this table in the model for time series analysis.

Date Dimension Table

Every Time creating this date fields for different data sets is difficult, so what we do is we create a Date Dimension Table which contains all these fields and use it.

DAX CALENDAR Function

The calendar function returns a table with a single column that contains a continuous set of dates. The start and end date range will be supplied as parameters.
The following formula returns a calculated table **Date_Dim** with dates between January 1st, 2011 and December 31st, 2020.

Date_Dim = CALENDAR (DATE (2011, 1, 1), DATE (2020, 12, 31))

From the above **Date** Column, we will be deriving below columns.

- Day = DAY(Date_Dim[Date])
- WeekDay = WEEKDAY(Date_Dim[Date])
- WeekDayName = FORMAT(Date_Dim[Date],"DDDD")
- Month = MONTH(Date_Dim[Date])
- MonthName = FORMAT(Date_Dim[Date],"MMMM")
- Quarter = ROUNDUP((Date_Dim[Month]/3),0)
- Quarter = IF (Month Number<4,"Qtr 1”, IF (Month Number<7,"Qtr 2”, IF (Month Number<10,"Qtr 3","Qtr 4")))
- QuarterName = "Qtr" & Date_Dim[Quarter]
- QuarterName = CONCATENATE ("Qtr", Date_Dim[Quarter])
- Year = YEAR(Date_Dim[Date])
- WeekNumber = WEEKNUM(Date_Dim[Date])
- WeekNumMonth = 1 + WEEKNUM (Dim_Date[Date]) - WEEKNUM(STARTOFMONTH(Dim_Date[Date]))
- WeekNameMonth = "Week" & " " & Date_Dim[WeekNumMonth]

Week of Month

The week number of the month is, one plus the difference between the weeknum for the date and the weeknum of the first of the month.

WeekOfMonth =1+ WEEKNUM (Dim_Date[Date]) - WEEKNUM(STARTOFMONTH(Dim_Date[Date]))

EOMONTH

EOMONTH Returns the date in datetime format of the last day of the month, before or after a specified number of months. Use EOMONTH to calculate maturity dates or due dates that fall on the last day of the month.

Syntax

EOMONTH (<start_date>, <months>)

Example

1. If customer taken a loan for 12 months, to find when the loan will complete use below syntax.

EOM = EOMONTH (Orders [Order Date], 12)

2. Based on Order Date filed in Orders table, get the Number of Days in that month.

No of Days in Month = DAY ((EOMONTH (Orders [Order Date], 12)))