Tasks

Another advantage of Power BI is how easily time intelligence can be added to your data model. DAX has a comprehensive list of built-in time intelligence functions that can be easily leveraged and add significant analytical value to your data model.

Take a look at the alternative methods for calculating time intelligence in the DAX cheat sheet at https://tinyurl.com/pbiqs-daxcheatsheet.

In order to leverage time intelligence functions in DAX, you must have a date table in your data model and that date table must have all available dates with no gaps. These are very important conditions that must be met. Oftentimes, developers try to use the date column from their transaction table (fact table). This can result in calculations that do not work and return incorrect results.

Task 1 - Create year-to-date calculations

Step 1: Create a new calculated measure in your *Internet Sales* table using the following DAX formula:

```
YTD Sales = TOTALYTD([Total Sales], 'Date (Order)'[Date])
```

Step 2: Format the measure as *$ English (United States)*.

Maybe your requirement is slightly more complex, and you need to see the year-to-date sales based on your fiscal year end rather than the calendar year-end date. The TOTALYTD function has an optional parameter that allows you to change the default year-end date from 12/31 to a different date.

Step 3: Create a new calculated measure in your *Internet Sales* table using the following DAX formula:

```
Fiscal YTD Sales = TOTALYTD([Total Sales], 'Date (Order)'[Date], "03/31")
```

Step 4: Show both of these new measures in a table in Power BI:

Year	English Month Name	Total Sales	YTD Sales
2005	July	$473 388,16	$473 388,16
2005	August	$506 191,69	$979 579,85
2005	September	$473 943,03	$1 453 522,89
2005	October	$513 329,47	$1 966 852,36
2005	November	$543 993,41	$2 510 845,77
2005	December	$755 527,89	$3 266 373,66
2006	January	$596 746,56	$596 746,56
2006	February	$550 816,69	$1 147 563,25
2006	March	$644 135,20	$1 791 698,45
2006	April	$663 692,29	$2 455 390,74
2006	May	$673 556,20	$3 128 946,94
Total		**$29 358 677,22**	

Year	English Month Name	Total Sales	Fiscal YTD Sales
2005	July	$473 388,16	473 388,16
2005	August	$506 191,69	979 579,85
2005	September	$473 943,03	1 453 522,89
2005	October	$513 329,47	1 966 852,36
2005	November	$543 993,41	2 510 845,77
2005	December	$755 527,89	3 266 373,66
2006	January	$596 746,56	3 863 120,21
2006	February	$550 816,69	4 413 936,91
2006	March	$644 135,20	5 058 072,11
2006	April	$663 692,29	663 692,29
2006	May	$673 556,20	1 337 248,48
Total		**$29 358 677,22**	

The newly created measures YTD Sales and Fiscal YTD Sales have both been added to the preceding tables. Let's take a closer look at how these two measures are different; the relevant sections in the table are annotated with the numbers one to four, corresponding to the following notes:

1. The amount displayed for December 2005 is $3,266,374.66 This is the cumulative total of all sales from January 1, 2005 to December 31, 2005.
2. As expected, the cumulative total starts over as the year switches from 2005 to 2006; therefore, the YTD Sales amount for January 2006 is $596,747.
3. In the Fiscal YTD Sales column, the cumulative total works slightly differently. The displayed amount of $5,058,072.11 is the cumulative total of all sales from April 1, 2005 to March 31, 2006.
4. Unlike the YTD Sales measure, the Fiscal YTD Sales measure does not start over until April 1. The amount displayed for April 2006 of $663,692.29 is the cumulative total for April. This number will grow each month until March 31, at which point, the number will reset again..

Task 2 - Create prior year calculations with CALCULATE()

A lot of time series analysis consists of comparing current metrics to the previous month or the previous year. There are many functions in DAX that work in conjunction with the CALCULATE function to make these types of calculations possible.

Step 1: To create a new measure to return the total sales for the prior year, create a new calculated measure in your *Internet Sales* table for **Prior Year Sales**:

```
Prior Year Sales =
CALCULATE(
    [Total Sales],              // SUM('Internet Sales'[Sales Amount]
    SAMEPERIODLASTYEAR(         // Change the filter context to go back one year
        'Date (Order)'[Date]))
```

Step 2: Show these new measures in a table in Power BI:

Year	English Month Name	Prior Year sales
2006	August	518 096,42
2006	December	714 476,19
2006	January	160 786,33
2006	July	349 490,19
2006	November	532 655,00
2006	October	509 319,41
2006	September	481 550,12
2007	April	621 103,13
2007	August	522 942,41
2007	December	558 568,72
2007	February	504 355,66
2007	January	597 207,21
Total		**29 358 677,22**

CALCULATE allows you to ignore or even change the current filter context. In the preceding formula, CALCULATE is used to take the current filter context and modify it to one year prior. This calculated measure also works at the day, month, quarter, and year levels of the hierarchy. For example, if you were looking at sales for June 15, 2020, then the Prior Year Sales measure would return sales for June 15, 2019. However, if you were simply analyzing your sales aggregated at the month level for June 2020, then the measure would return the sales for June 2019.

For a comprehensive list of all the built-in time intelligence functions, please take a look at https://tinyurl.com/pbiqs-timeintelligence.

End-of-Exercise

Olet suorittanut 0 % oppitunnista

◄ Exercise 13 - Understanding filter contex

Siirry...