Calculate with and without a filter

The way filters work in your report

Hello! I’m...

mahsasalimi

Data Enthusiast | BI Developer | Insight Creator

Come view my profile to learn more about my journey and how I can help your organization thrive!

578 1K 5K

Without Using FILTER

When you write an expression without explicitly using the *FILTER* function, DAX automatically evaluates the filter internally.

```
Professional Sales Without Filter =
CALCULATE([Total Sales],
    'Customers'[Occupation] = "Professional")
```

```
Professional Sales Without Filter - Real Operation =
CALCULATE( [Total Sales] ,
    FILTER( ALL( 'Customers'[Occupation] ) ,
        'Customers'[Occupation] = "Professional" ) )
```

What does it mean?

This behavior means that DAX applies the filter on the "*Professional*" occupation, overriding all other filters in the current context.

For example, even if you select another occupation using a slicer, the result will still display sales for the customers with "*Professional*" occupation.

The use of *ALL()* effectively removes other occupation filters, forcing the visual to show only the "*Professional*" data.

With Using FILTER

This approach forces to filter only the rows that match *[Occupation] = "Professional"*. While this works, it's not optimal because scanning an entire table can slow down performance, especially with larger datasets.

```
Professional Sales With Filter =
CALCULATE( [Total Sales] ,
    FILTER( 'Customers' ,
        'Customers'[Occupation] = "Professional" ) )
```

A better approach

By using KEEPFILTERS(), the engine avoids the full table scan and respects any existing filters applied in the visual. This ensures that the filter is applied without removing other filters already in place, leading to more efficient calculations and optimized performance.

```
Professional Sales - Keep filter =
CALCULATE([Total Sales],
    KEEPFILTERS('Customers'[Occupation] = "Professional"))
```