Filter

Tiene un máximo de 2 argumentos

La sintaxis de FILTER es la siguiente:

```
FILTER ( , <condition> )
```

FILTER se utiliza generalmente como una subfunción dentro de otra función, como SUMX o CALCULATE en una medida o para el cálculo de una columna calculada.

= FILTER() se utiliza para el cálculo de tablas calculadas, pero no, para crear medidas.

FILTER recibe una tabla y una condición lógica como parámetros. Como resultado, FILTER devuelve todas las filas que cumplen la condición. **FILTER es una función de tabla y un iterador al mismo tiempo**. Para devolver un resultado, escanea la tabla evaluando la condición fila por fila. En otras palabras, itera la tabla.

1. FILTER (Tabla calculada)

Por ejemplo, la siguiente <u>tabla calculada</u> devuelve los productos Fabrikam (siendo Fabrikam una marca).

```
FabrikamProducts =
FILTER (
    'Product',
    'Product'[Brand] = "Fabrikam"
)
```

2. FILTER (Tabla calculada)

Podemos crear una tabla calculada que contenga todos los productos con un precio unitario superior a 3000 utilizando una expresión de tabla como la siguiente:

```
ExpensiveProducts =
FILTER (
    'Product',
    'Product'[Unit Price] > 3000
)
```

3. SUMX / FILTER / RELATED (Medida)

FILTER se usa a menudo para reducir el número de filas en iteraciones. Por ejemplo, si un desarrollador desea calcular las ventas de productos rojos, puede crear una **medida** como la siguiente:

```
RedSales :=

SUMX (

FILTER (

Sales,

RELATED ('Product'[Color]) = "Red"

),

Sales[Quantity] * Sales[Net Price]

FILTER agrega una condición a las condiciones existentes. Por ejemplo,

RedSales en la fila Audio muestra las ventas de productos que son tanto de la categoría Audio como de color rojo.
```

Category	Sales Amount	RedSales
Audio	384,518.16	33,123.82
Cameras and camcorders	7,192,581.95	1,514.39
Cell phones	1,604,610.26	38,227.47
Computers	6,741,548.73	240,222.29
Games and Toys	360,652.81	19,938.31
Home Appliances	9,600,457.04	770,373.33
Music, Movies and Audio Books	314,206.74	6,702.49
TV and Video	4,392,768.29	
Total	30,591,343.98	1,110,102.10

4. FILTER / FILTER (Tabla calculada)

Es posible anidar FILTER en otra función de FILTER. En general, anidar dos filtros produce el mismo resultado que combinar las condiciones de dos funciones FILTER con una función AND. En otras palabras, las dos consultas siguientes producen el mismo resultado:

5. FILTER / AND (Tabla calculada)

```
FabrikamHighMarginProducts =
FILTER (
    'Product',
    AND (
        'Product'[Brand] = "Fabrikam",
        'Product'[Unit Price] > 'Product'[Unit Cost] * 3
    )
)
```

6. COUNTROWS / FILTER / (Medida)

Por ejemplo, imagina que necesitas calcular la cantidad de productos rojos

```
NumOfRedProducts :=
COUNTROWS (
    FILTER ( 'Product', 'Product'[Color] = "Red" )
)
```

7. COUNTROWS / FILTER (Medida)

Imagine que queremos crear una medida que cuente la cantidad de productos rojos. Con los conocimientos adquiridos hasta ahora, la fórmula es sencilla:

```
NumOfRedProducts :=
VAR RedProducts =
   FILTER (
         'Product',
         'Product'[Color] = "Red"
   )
RETURN
   COUNTROWS ( RedProducts )
```

Podemos usar esta fórmula dentro de un informe. Por ejemplo, coloque la marca del producto en las filas para generar el informe que se muestra a continuación:

Brand	NumOfRedProducts
Adventure Works	6
Contoso	36
Fabrikam	12
Litware	12
Northwind Traders	3
Proseware	7
Southridge Video	13
Tailspin Toys	6
Wide World Importers	4
Total	99

En el informe de la derecha, la tabla *Product* iterada por FILTER es filtrada por el color Azure y, como FILTER solo puede devolver productos Red, no hay productos para devolver. Como resultado, la medida *NumOfRedProducts* siempre se evalúa en blanco.

Azure	Brand	NumOfRedProducts	Color Azure Black Blue	Brand NumOfRedProducts
	Adventure Works	6		Total
	Contoso	36		
	Fabrikam	12	☐ Brown ☐ Gold	
	Litware	12	☐ Green	
	Northwind Traders	3	☐ Grey	
	Proseware	7	☐ Orange ☐ Pink	
	Southridge Video	13	☐ Purple	
	Tailspin Toys	6	☐ Red	
	Wide World Importers		☐ Silver ☐ Silver Grey	
	Total	99	☐ Transparent	
			☐ White	

La parte importante de este ejemplo es el hecho de que, en la misma fórmula, hay un contexto de filtro que proviene del exterior (la celda del informe, que se ve afectada por la selección de la segmentación) y un contexto de fila introducido en la fórmula por el Función FILTER. Ambos contextos funcionan al mismo tiempo y modifican el resultado. DAX utiliza el contexto de filtro para evaluar la tabla *Product* y el contexto de fila para evaluar la condición de filtro fila por fila durante la iteración realizada por FILTER.

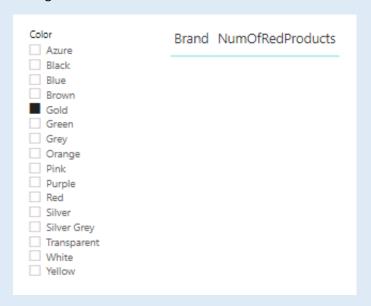
Queremos repetir este concepto nuevamente: FILTER no cambia el contexto del filtro. FILTER es un iterador que escanea una tabla (ya filtrada por el contexto del filtro) y devuelve un subconjunto de esa tabla, de acuerdo con la condición de filtrado. En la Figura 4-14, el contexto del filtro está filtrando la marca y, después de que FILTER devolvió el resultado, todavía solo filtró la marca. Una vez que agregamos la segmentación en el color de la Figura 4-15, el contexto del filtro contenía tanto la marca como el color. Por este motivo, en el informe del lado izquierdo FILTER devolvió todos los productos iterados, y en el informe del lado derecho no devolvió ningún producto. En ambos informes, FILTER no cambió el contexto del filtro. FILTER solo escaneó una tabla y devolvió un resultado filtrado.

Lo último se reduce a que:

Existen 2 FILTROS: Brand y el segmentador "Color".

Color Azure	Brand	${\bf NumOf Red Products}$
Black	A. Datum	4
Blue Brown	Adventure Works	12
Gold	Contoso	45
Green	Fabrikam	32
Grey Orange	Litware	31
Pink	Northwind Traders	8
Purple	Proseware	8
Red Silver	Southridge Video	6
Silver Grey	Tailspin Toys	39
Transparent	Wide World Importers	15
White Yellow	Total	200

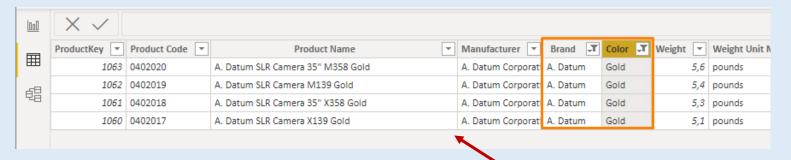
Si elegimos el Filtro del Color "Gold" nos devuelve En Blanco.



Esto sucede por:

1.- Se evalúa el contexto de filtro en la tabla Product.

2.- Por ejemplo, si nos fijamos en la marca "A. Datum". Dado que existen 2 FILTROS, la tabla Producto se verá filtrada por la MARCA y por el segmentador COLOR.



3.- Luego se aplicará el contexto de fila de FILTER, que iterará **esta tabla y buscará registros que** tengan el campo Color = "Blue". Dado que no encontrará nada, devolverá En blanco.

8. COUNTROWS / FILTER / ALL / (Medida)

En este punto, <u>es posible que desee definir otra fórmula que devuelva el número de productos rojos independientemente de la selección realizada en la segmentación</u>. En otras palabras, el código debe ignorar la selección realizada en la segmentación y siempre debe devolver el número de todos los productos rojos.

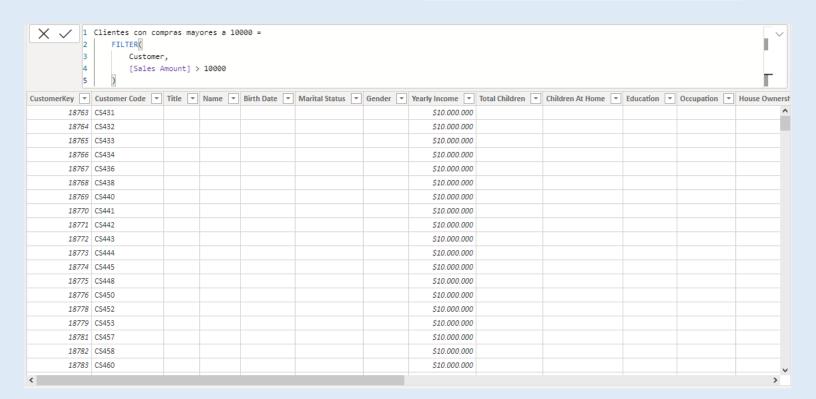
Para lograr esto, la función ALL es útil. ALL devuelve el contenido de una tabla ignorando el contexto del filtro. Podemos definir una nueva medida, llamada *NumOfAllRedProducts*, usando esta expresión:

```
NumOfAllRedProducts :=
VAR AllRedProducts =
   FILTER (
        ALL ( 'Product' ),
        'Product' [Color] = "Red"
    )
RETURN
   COUNTROWS ( AllRedProducts )
```

ALL ignora el contexto del filtro y siempre devuelve todas las filas de la tabla, de modo que FILTER devuelve los productos rojos incluso si los productos fueron previamente filtrados por otra marca o color.

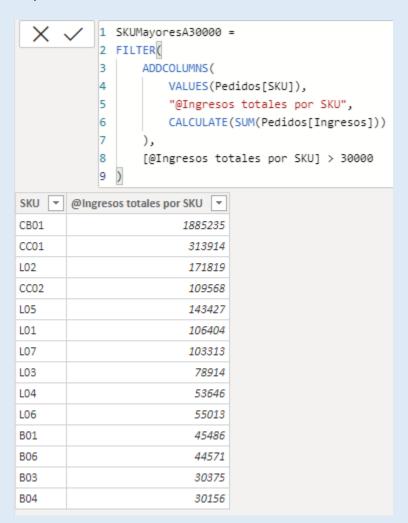
Azure Black Blue Brown Gold Green Grey Orange Pink Purple Red Silver Grey	Brand	NumOfAllRedProducts	Color Azure	Brand	NumOfAllRedProducts
	Adventure Works	99	☐ Black	A. Datum	99
	Contoso	99	Blue Total Brown Gold Green Grey Orange Pink Purple Red Silver	Total	99
	Fabrikam	99			
	Litware	99			
	Northwind Traders	99			
	Proseware	99			
	Southridge Video	99			
	Tailspin Toys	99			
	Wide World Importers	99			
	Total	99	☐ Transparent		
			☐ White		

9. FILTER / CALCULATE / SUMX (Tabla calculada)



10. FILTER / ADDCOLUMNS / VALUES (Tabla calculada)

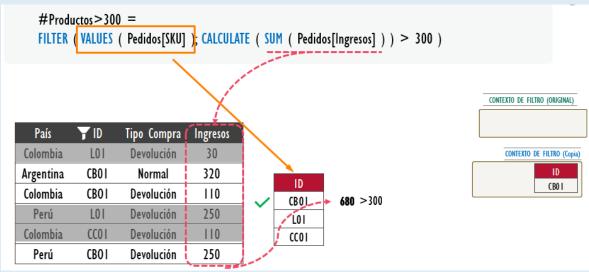
SKU con ingresos mayores a 30.000.

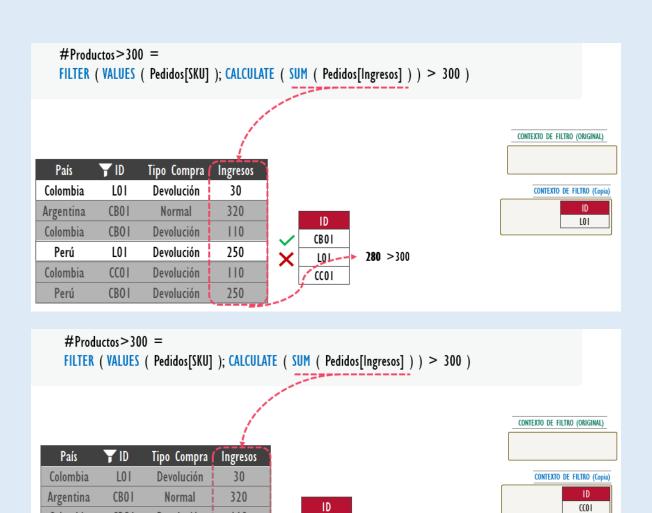


11. FILTER / VALUES (Tabla calculada)

SKU con ingresos mayores a 30.000.







CBOI

LOI

CCOI

110 >300

ID

CBOI

Colombia

Perú

Colombia

Perú

CBOI

LOI

CCOI

CBOI

Devolución

Devolución

Devolución

Devolución

110

250

110

250

12. FILTER / ADDCOLUMNS / VALUES (Tabla calculada)

Numero de filas de un producto especifico

```
SKU NumeroFilas CB01

1 SKUFiltroCB01 = 2 FILTER( 3 ADDCOLUMNS( 4 VALUES(Pedidos[SKU]), "NumeroFilas", 6 CALCULATE(COUNTROWS(Pedidos)) 7 ), 8 Pedidos[SKU] = "CB01" 9 )

SKU NumeroFilas CB01 4756
```

13. FILTER / ADDCOLUMNS / VALUES (Tabla calculada)

Total de ingresos por año



