**Funciones de filtro en CALCULAR**

**Introducción**

Las funciones de filtro son un elemento central del proceso de análisis de datos en DAX y Power BI. Estas funciones permiten a los analistas manipular los datos aplicando filtros a tablas y columnas, lo que permite un análisis de datos más preciso y granular. Las funciones de filtro se utilizan normalmente dentro de la función **CALCULAR** de DAX.

En esta lectura, explorará algunos ejemplos comunes de funciones de filtro utilizadas en **CALCULAR**, como las expresiones booleanas y de filtro de tabla. También revisará algunos ejemplos de funciones de modificación de filtros.

**Sintaxis**

Recapitulemos brevemente la sintaxis de **CALCULAR** con el siguiente ejemplo:

1

CALCULATE(<expression>[, <filter1> [, <filter2>])

La sintaxis de esta fórmula DAX comienza con la palabra clave **CALCULAR**, seguida del argumento entre paréntesis. El argumento contiene una expresión y uno o más filtros. Estos filtros son uno de los elementos clave de la función **CALCULAR**. Como ha aprendido anteriormente en este curso, los filtros pueden utilizarse para recuperar mayores niveles de granularidad de sus datos.

Repasemos los distintos tipos de filtros y cómo funcionan dentro de la función **CALCULAR**.

**Expresiones de filtro booleanas**

Una expresión de filtro booleana se evalúa como **VERDADERO** o **FALSO**. En otras palabras, la expresión devuelve un valor positivo o negativo.

Existen varias reglas que deben cumplir las expresiones de filtro booleanas:

* Pueden hacer referencia a columnas de una única tabla.
* No pueden hacer referencia a medidas.
* No pueden utilizar una función **CALCULAR** anidada.
* No pueden utilizar funciones que exploren o devuelvan una tabla, incluidas las funciones de agregación.

Por ejemplo, Adventure Works desea calcular las ventas totales de productos de gama alta, definidos como productos cuyo precio unitario es igual o superior a **500 $**. Este cálculo se puede realizar utilizando un filtro booleano con **CALCULAR** en DAX como se indica a continuación:

1

2

3

4

5

Sales of high-end products =

CALCULATE (

    SUM ( Sales[Total Sales] ),

    FILTER ( Products, Products[Unit Price] >= 500 )

)

La medida hace uso de una función **FILTRO** para aplicar su segundo argumento. Este argumento comprueba la columna **Precio unitario** de la tabla **Productos** y evalúa todos los registros como Verdadero o Falso. Todos los registros con un valor superior o igual a **500 $** se evalúan como **VERDADERO**. Por lo tanto, esta fórmula filtra todos los registros **TRUE**y devuelve una tabla que enumera todos los productos cuyo precio unitario es mayor o igual a 500 **$**.

**Expresiones de filtro de tabla**

Las expresiones de filtro de tabla implementan una tabla como filtro. Una expresión de este tipo podría hacer referencia a una tabla del modelo de datos, pero lo más probable es que sea una función que devuelva un objeto de tabla.

También puede utilizar la función**FILTRO** para aplicar condiciones de filtro complejas, incluidas aquellas que una expresión de filtro booleana no puede definir.

Por ejemplo, Adventure Works quiere generar sus ventas totales en **Alemania** para 2019. Para calcular esta fórmula, Adventure Works debe utilizar dos filtros de tabla. El primer filtro es de la tabla **Región**, y el segundo es de la tabla **Fecha**, como se indica a continuación:

1

2

3

4

5

2019 Sales in Germany =

CALCULATE (

    SUM ( [Total Sales] ),

    FILTER ( Region[Country] = "Germany", Date[Year] = "2019" )

)

En esta sentencia **CALCULAR**, la función **FILTRO** identifica todas las instancias de **Alemania** en la columna **País** de la tabla **Región**. También identifica todas las instancias de **2019** en la columna **Año** de la tabla **Fecha**. A continuación, devuelve estos resultados como una tabla calculada que calcula las ventas totales de Alemania en 2019.

**Funciones modificadoras de filtro**

Las funciones de modificación de filtros se utilizan para algo más que para añadir filtros. Proporcionan un control adicional al modificar un contexto de filtro. Por ejemplo, cuando añade un filtro a la sentencia **CALCULAR**, **CALCULAR** anula cualquier filtro existente creado previamente en la columna. Debe utilizar funciones modificadoras de filtro para asegurarse de que **CALCULAR** añade el nuevo filtro a cualquier filtro anterior.

He aquí un resumen de algunas de las funciones modificadoras de filtro más comunes utilizadas dentro de la función **CALCULAR** de DAX.

**REMOVEFILTERS**

La función **REMOVEFILTERS** se utiliza para eliminar todos los filtros o eliminar los filtros de una o varias columnas de una tabla o de todas las columnas de una única tabla.

Por ejemplo, se ha aplicado un filtro de **Producto** y **Región** a las tablas de la base de datos deAdventure Works.Usted desea analizar las ventas totales de la empresa sin ningún filtro aplicado a la tabla **Ventas**. Puede escribir una expresión DAX como la siguiente:

1

2

3

4

5

6

7

Total Sales =

CALCULATE

(

    [Total Sales],

    REMOVEFILTERS ( Product ),

    REMOVEFILTERS ( Region)

)

**REMOVEFILTERS** elimina todos los filtros de las tablas especificadas dentro de los paréntesis. En este caso, elimina los filtros de las dimensiones **Producto** y **Región**.

**KEEPFILTERS**

**KEEPFILTERS** añade un filtro sin eliminar los filtros existentes en las mismas columnas. En otras palabras, puede mantener los filtros existentes y seguir evaluando la expresión.

Por ejemplo, Adventure Works debe calcular sus ventas totales manteniendo el filtro de color existente en la tabla **Producto** . La expresión DAX necesaria es la siguiente

1

2

3

4

5

6

Blue Products Sale =

CALCULATE

(

    [Total Sales],

    KEEPFILTERS ( Products[Color] = "Blue" )

)

Si existe un filtro activo en **Product[Color]**, **KEEPFILTERS** se asegura de que el filtro no se anule. En su lugar, se fusiona con el nuevo filtro.

**ALL**

La familia de funciones **ALL(ALLEXCEPT, ALLSELECTED, ALLNOBLANKROW**) elimina los filtros de una o varias columnas. También pueden utilizarse para eliminar todas las columnas de una misma tabla.

Esta función actúa de forma equivalente a **REMOVEFILTERS**.

Por ejemplo, Adventure Works puede utilizar la función **ALL** para calcular las ventas totales de cada región de la siguiente manera:

1

2

3

4

5

6

7

Total Sales =

CALCULATE

(

    [Total Sales],

    ALL ( Product ),

    ALL ( Region)

)

En esta fórmula, la función **ALL** devuelve los resultados requeridos como una tabla o una columna, que puede utilizarse en otras funciones DAX. **REMOVEFILTERS** no devuelve una tabla o columna. En su lugar, sólo elimina los filtros de las tablas o columnas especificadas.

**CROSSFILTER**

La función **CROSSFILTER** modifica la dirección del filtro (de **Both** a **Single**, o de **Single** a **Both**). También puede utilizarse para desactivar una relación entre tablas.

Adventure Works desea analizar su número total de productos por un año específico. Sin embargo, la dirección de filtrado cruzado por defecto del modelo de datos no permite a la empresa calcular el valor. Por lo tanto, Adventure Works puede crear una medida utilizando DAX sin cambiar la dirección del filtro cruzado por defecto en el modelo de datos como se indica a continuación:

1

2

3

4

5

6

7

8

9

Product by Year =

CALCULATE

(

    DISTINCTCOUNT ( Products[ProductKey] ),

    CROSSFILTER

        ( Sales[ProductKey],

        Products[ProductKey],

        BOTH )

)

La fórmula analiza la medida basándose en la columna **Año** de la tabla **Fecha** y devuelve resultados precisos según las necesidades analíticas de la empresa.

**USERELATIONSHIP**

La función **USERELATIONSHIP** establece una relación inactiva entre columnas relacionadas. Esto significa que la relación activa se convertirá automáticamente en inactiva.

Por defecto, Power BI utiliza la relación activa para todos los análisis y visualizaciones. Sin embargo, puede haber ocasiones en las que su análisis requiera utilizar la relación inactiva. En este caso, puede utilizar **la** relación activa.

Por ejemplo, Adventure Works quiere calcular sus ventas totales utilizando la columna **Fecha de envío** de la tabla **Ventas** y la columna **Fecha** de la tabla **Fecha**. Esta relación está actualmente inactiva. Entonces, Adventure Works puede utilizar la función **USERELATIONSHIP** dentro de la función **CALCULAR** de la siguiente manera:

1

2

3

4

5

Sales by Shipping date =

CALCULATE (

    SUM ( Sales[SalesAmount] ),

    USERELATIONSHIP ( Sales[ShippingDate], Date[Date] )

)

En esta fórmula, la función **USERELATIONSHIP** sólo cambia la relación por defecto entre las tablas relacionadas para las medidas actuales. Recuerde que este enfoque suele manejar las relaciones inactivas dentro de las dimensiones de los juegos de rol.

**Puntos a recordar**

**Utilización de operadores lógicos**

Cuando utilice varios filtros en la función **CALCULAR**, pueden ejecutarse utilizando operadores lógicos.

Puede utilizar el operador lógico **AND(&&)**, que significa que todas las condiciones deben ser verdaderas. O puede utilizar el operador lógico **OR(||**), lo que significa que cualquiera de las condiciones puede ser verdadera. El operador lógico que utilice dependerá de sus requisitos analíticos.

**Funciones de modificación del filtro ALL**

Las funciones de modificación del filtro **ALL** funcionan como modificadores del filtro y como una función que devuelve un objeto de tabla.

**CALCULAR los resultados del filtro**

Los resultados de un argumento de filtro en la función **CALCULAR** se presentan siempre como una tabla con una o varias columnas. Puede utilizar una función de filtro dentro de la función **CALCULAR** para crear tablas virtuales de forma dinámica basándose en las condiciones del filtro. El resultado del argumento del filtro es siempre una tabla virtual con una o más columnas. El filtro reduce el número de filas de la tabla.

**Conclusión**

Los filtros son un potente conjunto de funciones que se utilizan dentro de la sentencia **CALCULAR** para lograr los requisitos analíticos de la empresa. Es importante conocer a fondo las funciones y los modificadores de filtro para liberar todo el potencial de **CALCULAR** para cambiar el contexto de evaluación de los cálculos.