

Función TREATAS

Compatibilidad

Microsoft Excel
Excel ≥ 2019

★★★★★

 Se puede usar desde Excel 2013/1809, pero no aparece en IntelliSense. ¹

Power BI Desktop
PBI ≥ Feb 2017

★★★★★

SQL Analysis Services
SSAS ≥ 2017

DIRECTQUERY: C.Calculadas Medidas
ROW LEVEL SECURITY:

Int. Contexto

Contexto de Filtro
Tiene en cuenta el contexto de filtro

★★★★★

Contexto de Fila

Categorías

Según Proceso Interno
Indeterminado

★★★★★

Según Resultado
Tabla

Recursos de Aprendizaje



MAGÍSTER EN LEGUAJE DAX

100% en Vivo - Más Información:

→ [Capacitación OnLine] ←

<https://bit.ly/3bz1kG0>



TREATAS: Justiciero del Data Lineage

Descripción

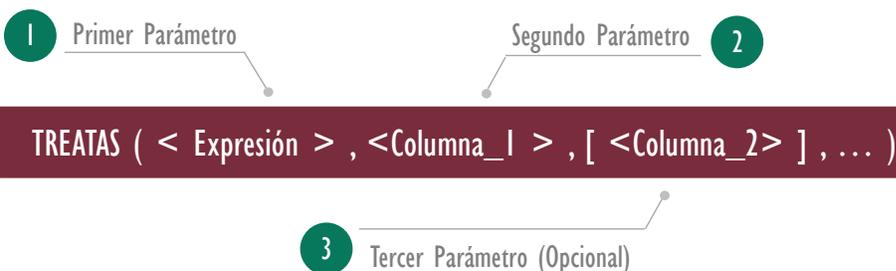
La función **TREATAS** es descrita a menudo como una función que trata las columnas de una tabla creada programablemente o cargadas de forma externa como si fuera las columnas de una tabla nativa.

- De allí su nombre: **TREAT**: Tratar esta columna, **AS**: Como filtro para otra.

Su descripción oficial busca que entendamos lo que hace de una forma rápida, certera e intuitiva, no obstante, no debemos perder de vista su especificación interna: **TREATAS** es una función que actualiza o hereda el *data lineage* (linaje) de columnas de tablas nativas a columnas de una tabla creada programablemente o cargada de forma externa, significando que: aplica los resultados de una expresión como filtros de columnas a una tabla no relacionada.

- **Data Lineage (Linaje de datos)**: se puede visualizar como una etiqueta especial añadida a cada columna, la cual identifica su posición de manera única en el modelo de datos, estas se pueden heredar.

Sintaxis



1 Expresión

La expresión de tipo tabla o tabla externa no relacionada a ser remapeada en su etiqueta.

① Expresión Tabular

2 Columna_1

Columna nativa en el modelo (generalmente relacionada) que heredará su data lineage a la primera columna de la expresión tabular.

Tipo

Obligatorio

Atributo

No Repetible 

Tipo

Obligatorio

Atributo

Repetible 

Más Recursos de Aprendizaje



Máster en DAX y Power Pivot

Tomo Número 7: Módulo 22

→ [Visitar Curso] ←

El curso **Máster en DAX y Power Pivot** es un vídeo curso pregrabado de 24 horas dedicado a explorar el lenguaje DAX y la interfaz de Power Pivot en Excel.

A lo largo de sus 164 lecciones distribuidas en 22 módulos **se abraza las mieles del análisis de datos para disfrutar de las bondades de la creación de KPIs y KRIs** de todo tipo utilizando las funciones DAX y tablas dinámicas, cada archivo cuenta con la documentación completa del material tratado con expresiones DAX y explicación.

Es una versión austera del más completo **Magíster en Lenguaje DAX: Nivel 1, 2 y 3** (dictado 100% en directo) pero ideal para usuario de Excel que utilizan lenguaje DAX con Power Pivot.

→ <https://bit.ly/3wLQ0Rd> ←



TREATAS: Justiciero del Data Lineage

APLICACIONES

La función **TREATAS** es empleada en por lo menos 4 ramas: *Creación de filtros arbitrarios*, *Optimización de medidas semi aditivas*, *Propagación de filtros para relaciones virtuales* y cómo *Parámetro de filtro en SUMMARIZECOLUMNS*, esto no condiciona que sean las únicas aplicaciones, pero si las más comunes.

- I. **Creación de Filtros Arbitrarios:** Un filtro arbitrario lista explícitamente el subconjunto de combinaciones que pueden ser visibles del total de posibilidades cuando es aplicado en el modelo de datos.

- Ejemplo:

En la compañía Disproductos L.D existen 6 categorías de prioridad de envíos que el cliente puede seleccionar para la entrega de su pedido: *Crystal* (1 día), *Platinum* (2 a 4 días), *Gold*, (1 a 4 semanas) *Silver* (1 a 2 meses), *Bronze* (2 a 4 meses) y *Iron* (4 a 8 meses).

Por distancia y procesamiento en aduanas existen dos combinaciones que requieren de acción inmediata para que se puede cumplir la entrega en el tiempo estipulado, específicamente: **Argentina/Crystal**, y, **Uruguay /Platinum** (*La combinación Uruguay – Crystal no es posible en la plataforma*). Estas combinaciones específicas de posibilidades se pueden presentar por diversidad de motivos, el ejemplo actual se basa en que, por un lado, la combinación *Argentina/Crystal* requiere acción inmediata por la duración de transporte (distancia), mientras que la combinación *Uruguay/Platinum* demanda acción inmediata por el tiempo de validación para entrada al país (aduanas).

Si se desea calcular el número de ventas que requieren acción inmediata, basta con contar el número de filas para las dos combinaciones señaladas, esto se puede conseguir sin **TREATAS**, así:

Videos Sobre TREATAS



Entendiendo el Linaje de datos (Ricardo Rincón)

En el seminario impartido en la **IV Jornada Internacional de Tecnología y Negocios** a disposición para ver en diferido en YouTube, el **MVP Ricardo Rincón** proporciona una descripción detallada de la función **TREATAS**, además de brindarnos una introducción a la función **CALCULATE**, donde luego nos enseña su utilidad de **TREATAS** y **CALCULATE** en aplicaciones utilizando propagación de filtros virtuales.

→ <https://bit.ly/3Roa3gs> ←



Usando la función TREATAS

En el video **Sandielly Ortega** brinda una descripción y ejemplos para conocer el comportamiento de la función **TREATAS**.

→ <https://bit.ly/3wNeKZo> ←



TREATAS: Justiciero del Data Lineage

```

1. #VtsAccnInmdta =
2. CALCULATE (
3.     COUNTROWS ( Pedidos ),
4.     FILTER (
5.         ALL ( Pedidos[País], Pedidos[Prioridad] ),
6.         ( Pedidos[País] = "Argentina"
7.           && Pedidos[Prioridad] = "Crystal" )
8.         || ( Pedidos[País] = "Uruguay"
9.            && Pedidos[Prioridad] = "Platinum" )
10.    ) -- Si el número de combinaciones requeridas es mayor, digamos:
11. ) -- 6, su lectura se hace más difícil.
    
```

Aunque la solución previa resuelve el escenario, podemos argüir dos inconvenientes: primero si la cardinalidad de las columnas es baja en contraste a una cantidad de registro notable, la expresión puede tener problemas de rendimiento, segundo, es viable escribirlas de una forma más elegante gracias **TREATAS**, que por añadidura mejora su rendimiento, así:

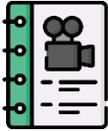
```

1. #VtsAccnInmdta =
2. CALCULATE (
3.     COUNTROWS ( Pedidos ),
4.     TREATAS (
5.         { ( "Argentina", "Crystal" ), ( "Uruguay", "Platinum" ) },
6.         Pedidos[País],
7.         Pedidos[Prioridad]
8.     ) -- Se ha simplificado la expresión de 11 líneas a 9 líneas
9. ) -- Más importante aún, es más fácil de leer.
    
```

II. **Optimización de Medidas Semi Aditivas:** Se puede ver que **TREATAS** es implementada en optimización de código DAX. Una de las aplicaciones básica más extendidas en este sentido viene de la mano con las medidas semi aditivas.

Para aprender más:  [Magister en Lenguaje DAX – Nivel I](#)

Otros Recursos de Aprendizaje



El ADN de Power Pivot

Capítulo número 8

→ [Visitar Libro] ←

El libro **El ADN de Power Pivot** es un manuscrito para estudiar los fundamentos y aspectos intermedios del lenguaje DAX, utilizando Excel y específicamente Power Pivot como herramienta cliente, para la creación de expresiones DAX cuyas soluciones se proyectan a través de tablas dinámicas, además, materializando tablas en la hoja de cálculo.

Es una primera guía para adentrarse en el mundo de DAX, sin embargo, si se requiere más información de la función TREATAS se debe optar por un vídeo curso como el [Máster en DAX y Power Pivot](#) o capacitaciones online y 100% en vivo como: [Énfasis en Contexto de Filtro](#).

→ <http://eladndepowerpivot.com/> ←

• Ejemplo

Una medida que retorne el último balance dado que los clientes tienen fechas distintas de registro final, por consiguiente, su balance es el último dato en el periodo, mientras que el balance total del “banco” es la sumatoria de los últimos balances de cada cliente:

```

1.  ÚltimoBalanceOptimizado =
2.  VAR UltimoClienteyFecha =
3.      ADDCOLUMNS (
4.          VALUES ( Balance[Nombre] ),
5.          "UltimaFecha", CALCULATE (
6.              MAX ( Balance[Fecha] ),
7.              DATESBETWEEN ( Calendario[Fecha],
8.                  BLANK (),
9.                  MAX ( Calendario[Fecha] ) )
10.         )
11.     )
12. )
13. VAR FiltrosDeCleintesyFechas =
14.     TREATAS ( UltimoClienteyFecha, Balance[Nombre], Calendario[Fecha] )
15. VAR SumaDelUltimoBalance =
16.     CALCULATE ( SUM ( Balance[Balance] ), FiltrosDeCleintesyFechas )
17. RETURN
18.     SumaDelUltimoBalance
    
```

III. **Propagación de Filtros para Relaciones Virtuales:** Debido a las relaciones físicas los filtros de una tabla afectan a otra y ello dota a DAX de una facilidad única para trabajar con múltiples tablas, a pesar de ello, existen escenarios donde crear la relación es complicado.

En primer lugar, puede ser demasiado costoso en términos de rendimiento crear una columna calculada con las columnas nativas concatenadas necesarias para crear la relación física. *En segundo lugar*, se puede emular la propagación de filtros muchos a muchos sin insertar ambigüedad en el modelo.



NOTA



La implementación de propagación de filtros virtual puede parecer poco común, pero si pensamos en:

TAMAÑOS DE GRANOS DIFERENTES



Donde el nivel de detalle en una tabla no coincide con el de otro, por ejemplo, en una tabla puede estar el detalle a nivel de días, incluso horas, mientras que en otra el detalle máximo se alcanza a nivel de meses, además si:

EXISTEN MÚLTIPLES COLUMNAS



La relación puede depender de múltiples columnas para crear las combinaciones únicas y el grado de detalle necesario, por ejemplo: una tabla con el importe de publicidad para los productos, donde se registra mes, monto, producto a publicitar y año.

Con [TREATAS](#) es posible emular la propagación de filtros privando al modelo de datos de la existencia física de la relación.

- Ejemplo:

La siguiente medida propaga los filtros de la tabla de dimensión *SKUProductos* a la tabla de hechos *Pedidos* estando ausente la relación vivida (física) en la interfaz.

```

1.  IngVrt =
2.  CALCULATE ( [Ing],
3.              TREATAS ( VALUES ( SKUProductos[SKU] ),
4.                      Pedidos[SKU]
5.                    )
6.  ) -- Propagación de filtro de un campo (una sola tabla).
    
```

Si se desea la propagación de filtros virtual de campos de tablas diferentes, la metodología óptima consiste en emplear [SUMMARIZE](#) para crear las agrupaciones, apoyándonos de una tabla de hechos diferente a la del cálculo objetivo donde si existen la relación física entre las tablas.

```

1.  IngVrtMultiplesTablas =
2.  CALCULATE (
3.      SUM ( Pedidos[Ingresos] ),
4.      TREATAS (
5.          SUMMARIZE ( Produccion, Calendario[Año], SKUproductos[SKU] ),
6.          Pedidos[Año],
7.          Pedidos[SKU]
8.      )
9.  ) -- Propagación de filtros de campos de múltiples tablas.
    
```

Cabe resaltar que las relaciones entre la tabla *Pedidos*, *SKUProductos* y *Calendario* no existen, empero, las relaciones físicas entre las tablas *Producción*, *SKUProductos* y *Calendario* si existen, por ende, [SUMMARIZE](#) puede crear las agrupaciones de las posibles combinaciones existentes.



NOTA



La propagación virtual no sólo es posible con TREATAS, de hecho, en versiones antiguas de DAX no existía.

PROPAGACIÓN CON INTERSECT



Una manera alterna de conseguirlo es con la ayuda de la función INTERSECT, que frente a la sintaxis de TREATAS es muy similar, pero siguiendo un orden diferente, **esto evidentemente necesita más desarrollo y profundización.**

PROPAGACIÓN CON FILTER



La opción menos recomendada por su bajo rendimiento, pero con la gran ventaja que puede ser implementada en cualquier versión de DAX.

Si deseas aprender más acerca de relaciones virtuales el [Tomo No. 7 del Máster en DAX y Power Pivot](#) es una opción.

Si la relación física entre una o más tablas no es viable con la tabla transaccional de apoyo para crear las agrupaciones con [SUMMARIZE](#), entonces, dicho filtro se debe indicar en su propio [TREATAS](#) en un parámetro de filtro aparte en [CALCULATE](#). [\[Véase la ficha técnica de CALCULATE\]](#).

Tal es el caso de la tabla de dimensión *Descuentos* con la tabla de hechos de *Producción*, debido a que la relación entre ellas no es factible, puesto que: *¿Qué tiene que ver el proceso de producción con descuento?*

Para esta situación podemos optar por una solución del siguiente estilo:

```

1.  IngVrtMCol =
2.  CALCULATE (
3.    [Ing],
4.    TREATAS (
5.      SUMMARIZE ( Producción, Calendario[Año], SKUproductos[SKU] ),
6.      Pedidos[Año],
7.      Pedidos[SKU]
8.    ),
9.    TREATAS (
10.     VALUES ( Descuentos[CategoriaDescuento] ),
11.     Descuentos[CategoriaDescuento]
12.   )
13. )
    
```

- IV. **Parámetros de Filtros para la Función SUMMARIZECOLUMNS:** La función [SUMMARIZECOLUMNS](#) es la función definitiva para crear consultas con el lenguaje DAX, puesto que combina las fuerzas de [SUMMARIZE](#), [ADDCOLUMNS](#) y [CALCULATETABLE](#) en un solo lugar, incluso está equipada con modificadores como [IGNORE](#), [ROLLUPADISSUBTOTAL](#) y [ROLLUPGROUP](#) para ignorar la omisión de resultados en blanco, agregar subtotales específicos o de grupos. Para más información: [\[Véase la ficha técnica de SUMMARIZECOLUMNS\]](#).



Una habilidad potente de [SUMMARIZECOLUMNS](#) consiste en inyectar filtros al contexto de filtro al estilo de [CALCULATE](#), sin embargo, estos filtros se deben indicar en su forma tabular de forma explícita y no como syntax sugar. [\[Véase la ficha técnica de CALCULATE\]](#).

La forma compacta y más fácil de generar estos *filtros (los filtros son tablas)* es con [TREATAS](#):

```
1. EVALUATE
2. SUMMARIZECOLUMNS (
3.     -- Tabla de resumen agrupada por el campo SKU
4.     SKUProductos[SKU],
5.     -- Filtro a inyectar en el contexto de filtro
6.     "CategoríaDeDescuento", TREATAS ( { "Cyber Monday" },
7.                                         Descuentos[Descuento]
8.                                         ),
9.     -- Columna que indique los ingresos
10.    "Ingresos", [Ing]
11. )
```



SUMMARIZECOLUMNS ES (CASI) SÓLO PARA CONSULTAS

No queremos dejar esta discusión sin resaltar que **las maravillas de SUMMARIZECOLUMNS no se pueden extrapolar para aplicar a medidas**, ya que viene sujeta a una gran limitación:

SUMMARIZECOLUMNS no puede ser llamada si un contexto de filtro externo ha activado la operación de transición de contextos.

Podría parecer que aun así tiene bastante implantación en medidas, pero **no nos dejemos engañar, cada vez que utilizamos una medida** por sencilla que sea, puede ser un *SUM (Pedidos[Ingresos])*, es para arrestarla a una *matriz, gráficos de barras*, etc. Y estos objetos visuales para su creación deben ejecutar una consulta a la base de datos, generando internamente un código DAX, que entre otras cosas viene acompañado de un [ADDCOLUMNS](#), función de iteración que crea un contexto de fila programable y activa transición de contexto en presencia de [CALCULATE](#).



Acerca de las Cartas DAX



Las cartas DAX del equipo de **Excel Free Blog** es un paquete de contenido de documentación y representación para un juego de todas las funciones en lenguaje DAX, compuesta por dos partes:

I. La Carta

Cada función en todo el lenguaje DAX contará con un **personaje representativo**, por ejemplo, la función SUMX será representada por el ser mitológico: el grifo.

II. La Ficha Técnica

La ficha técnica tiene **información de la función** para su manejo, consulta y entendimiento, en ella se documenta y explica: Descripción, sintaxis, parámetros y más. (Cómo la presente)

Más Información

→ <https://bit.ly/3aZiBqu> ←
→ www.CartasDax.Com ←

Última Actualización:

2 de septiembre del 2022.



TREATAS: Justiciero del Data Lineage

OBSERVACIONES

- I. El resultado de la función **TREATAS** se puede asignar a una variable y mantendrá el data lineage heredado, pues **TREATAS** no es un modificador de parámetros como **KEEPFILTERS**, en lugar de ello es una función que actualiza la etiqueta de posición de una o más columnas en el modelo de datos.
- II. La función **TREATAS** puede ser utilizada en Excel 2013 a partir de la actualización 1809, no obstante, no aparece listada en el menú desplegable de autocompletado (IntelliSense) y no existe soporte por parte de Microsoft esta función para Excel. ¹

Algunos entusiastas tienen como solicitud que se le de un tratamiento a la función **TREATAS** de manera normal en Power Pivot para Excel en la comunidad *Excel User Voice*.

BIBLIOGRAFÍA

Seminarios:

- BITODATA (Ricardo Rincón): Entendiendo el Linaje de datos
https://www.youtube.com/watch?v=IIHtECdWR_I

Páginas Web:

- 1. DAX GUIDE: <https://dax.guide/treatas/>
- 2. MICROSOFT: <https://docs.microsoft.com/en-us/dax/treatas-function>
- 3. EXCELERATOBI: <https://exceleratorbi.com.au/virtual-filters-using-treatas/>

Libros:

- Definitive Guide To DAX (2nd Edition) — Marco Russo y Alberto Ferrari 
- Exam Ref 70-778 Analyzing and Visualizing Data — Daniil Maslyuk 

Creado por:

Miguel Caballero y Fabian Torres.

Cualquier Retroalimentación:

excelfreebymcs@gmail.com

Funciones Relacionadas:  [INTERSECT](#)