

# Funciones de filtro (DAX)

## Función ALL

Devuelve todas las filas de una tabla o todos los valores de una columna; se omiten los filtros que se puedan haber aplicado. Esta función resulta útil para borrar filtros y crear cálculos en todas las filas de una tabla.

### Sintaxis

`ALL( {<table> | <column>[, <column>[, <column>[,...]]]} )`

### Parámetros

Término	Definición
table	La tabla de la que desea borrar filtros.
column	La columna de la que desea borrar filtros.

El argumento de la función ALL debe ser una referencia a una tabla base o a una columna base. Con la función ALL no se pueden usar expresiones de tabla o de columna.

### Valor devuelto

La tabla o la columna con los filtros quitados.

### Comentarios

Esta función no se usa por sí sola, sino que sirve de función intermedia que se puede usar para cambiar el conjunto de resultados en los que se realizan otros cálculos.

<Estado del tema: Parte de la información de este tema es una versión preliminar y está sujeta a cambios en versiones futuras. La información preliminar describe las nuevas características o cambios de las características existentes en Microsoft SQL Server 2014 Community Technology Preview 1 (CTP1). En los casos en los que [Column] se marca como columna Date con Mark como tabla de fecha

Tal y como se describe en la siguiente tabla, puede utilizar funciones ALL y ALLEXCEPT en escenarios diferentes.

Función y uso	Descripción
ALL(Table)	Quita todos los filtros de la tabla especificada. La función ALL(Table) devuelve todos los valores de la tabla y quita los filtros del contexto que, de otra forma, se podrían haber aplicado. Esta función resulta útil cuando se trabaja con muchos niveles de agrupación y se desea elaborar un cálculo que cree una proporción de un valor agregado al valor total. En el primer ejemplo se demuestra este escenario.
ALL (Column[, Column[, ...]])	Quita todos los filtros de las columnas especificadas de la tabla; todos los demás filtros de las demás columnas de la tabla se siguen aplicando. Todos los argumentos de columna deben proceder de la misma tabla. La variante ALL(Column) resulta útil cuando se desea quitar los filtros de contexto de una o varias columnas específicas, y conservar todos los demás filtros de contexto. En los ejemplos segundo y tercero se demuestra este escenario.

ALLEXCEPT(Table, Column1 [,Column2]...)	Quita todos los filtros de contexto de la tabla excepto los filtros que se aplican a las columnas especificadas. Es un acceso directo conveniente para las situaciones en las que desea quitar los filtros en muchas columnas de una tabla, pero no en todas.
---	--

## Ejemplo: calcular la proporción de ventas por categoría con respecto al total de ventas

### Descripción

Supongamos que desea averiguar el importe de ventas de la celda actual, en su tabla dinámica, dividido entre las ventas totales de todos los distribuidores. Para asegurarse de que el denominador es el mismo independientemente del modo en el que el usuario de la tabla dinámica pueda filtrar o agrupar los datos, defina una fórmula que use ALL para crear el total general correcto.

La tabla siguiente muestra los resultados cuando se crea la nueva medida, llamada All Reseller Sales Ratio, usando la fórmula de la sección de códigos. Para ver cómo funciona esto, agregue el campo CalendarYear al área **Etiquetas de fila** de la tabla dinámica, y agregue el campo ProductCategoryName al área **Etiquetas de columna**. A continuación, arrastre la medida, All Reseller Sales Ratio, al área **Valores** de la tabla dinámica. Para ver los resultados como porcentajes, utilice las características de formato de Excel para aplicar un formato de número de porcentaje a las celdas que contienen la medida.

All Reseller Sales	Etiquetas de columna				
Etiquetas de fila	Accessories	Bikes	Clothing	Components	Grand Total
2005	0.02%	9.10%	0.04%	0.75%	9.91%
2006	0.11%	24.71%	0.60%	4.48%	29.90%
2007	0.36%	31.71%	1.07%	6.79%	39.93%
2008	0.20%	16.95%	0.48%	2.63%	20.26%
Grand Total	0.70%	82.47%	2.18%	14.65%	100.00%

### Código

```
=SUMX(ResellerSales_USD,  
ResellerSales_USD[SalesAmount_USD])/SUMX(ALL(ResellerSales_USD),  
ResellerSales_USD[SalesAmount_USD])
```

### Comentarios

La fórmula se construye del siguiente modo:

1. El numerador, `SUMX(ResellerSales_USD, ResellerSales_USD[SalesAmount_USD])`, es la suma de los valores de ResellerSales\_USD[SalesAmount\_USD] para la celda actual de la tabla dinámica, con los filtros de contexto aplicados en CalendarYear y ProductCategoryName.
2. Para el denominador, se comienza mediante la especificación de una tabla, ResellerSales\_USD, y se usa la función ALL para quitar todos los filtros de contexto de la tabla.
3. A continuación, se usa la función SUMX que suma los valores de la columna ResellerSales\_USD[SalesAmount\_USD]. Es decir, se obtiene la suma de ResellerSales\_USD[SalesAmount\_USD] para todas las ventas de distribuidor.

Para obtener más información acerca de la creación de medidas, vea [Create a Measure](#).

Nota
------

En el ejemplo anterior se usan las tablas ResellerSales\_USD, DateTime y ProductCategory del libro de ejemplo de DAX. Para obtener más información acerca de los datos de ejemplo, vea [Obtener datos de ejemplo](#).

## Ejemplo: calcular la proporción de las ventas por producto a lo largo del año actual

### Descripción

Suponga que desea crear una tabla que muestre el porcentaje de ventas comparado a lo largo de los años por cada categoría de producto (ProductCategoryName). Para obtener el porcentaje de cada año sobre cada valor de ProductCategoryName, es necesario dividir la suma de ventas de ese año concreto y la categoría de producto entre la suma de ventas para la misma categoría de producto durante todos los años. Es decir, desea mantener el filtro de ProductCategoryName pero quita el filtro del año al calcular el denominador del porcentaje.

La tabla siguiente muestra los resultados cuando se crea la nueva medida, llamada Reseller Sales Year, usando la fórmula de la sección de códigos. Para ver cómo funciona esto, agregue el campo CalendarYear al área **Etiquetas de fila** de la tabla dinámica, y agregue el campo ProductCategoryName al área **Etiquetas de columna**. Para ver los resultados como porcentajes, utilice las características de formato de Excel para aplicar un formato de número del porcentaje a las celdas que contienen la medida Reseller Sales Year.

Reseller Sales Year	Etiquetas de columna				
Etiquetas de fila	Accessories	Bikes	Clothing	Components	Grand Total
2005	3.48%	11.03%	1.91%	5.12%	9.91%
2006	16.21%	29.96%	27.29%	30.59%	29.90%
2007	51.62%	38.45%	48.86%	46.36%	39.93%
2008	28.69%	20.56%	21.95%	17.92%	20.26%
Total general	100.00%	100.00%	100.00%	100.00%	100.00%

### Código

```
=SUMX(ResellerSales_USD, ResellerSales_USD[SalesAmount_USD])/CALCULATE(SUM(ResellerSales_USD[SalesAmount_USD]), ALL(DateTime[CalendarYear]))
```

### Comentarios

La fórmula se construye del siguiente modo:

1. El numerador, **SUMX(ResellerSales\_USD, ResellerSales\_USD[SalesAmount\_USD])**, es la suma de los valores de ResellerSales\_USD[SalesAmount\_USD] para la celda actual de la tabla dinámica, con los filtros de contexto aplicados en las columnas CalendarYear y ProductCategoryName.
2. Para el denominador, quite el filtro existente en CalendarYear utilizando la función ALL(Column). Esto calcula la suma sobre las filas restantes en la tabla ResellerSales\_USD, después de aplicar los filtros de contexto existentes desde las etiquetas de columna. El efecto neto es que, para el denominador, la suma se calcula sobre el valor de ProductCategoryName seleccionado (el filtro de contexto implícito) y para todos los valores de Year.

Para obtener más información acerca de la creación de medidas, vea [Create a Measure](#).

#### Nota

Este ejemplo usa las tablas ResellerSales\_USD, DateTime y ProductCategory del libro de ejemplo de DAX. Para obtener más información acerca de los datos de ejemplo, vea [Obtener datos de ejemplo](#).

# Ejemplo: calcular la contribución de las categorías del producto a las ventas totales por año

## Descripción

Suponga que desea crear una tabla que muestre el porcentaje de ventas para cada categoría de producto, en una base año a año. Para obtener el porcentaje de cada categoría de producto de un año determinado, es necesario calcular la suma de ventas para esa categoría de producto concreta (ProductCategoryName) en el año n, a continuación, dividir el valor resultante por la suma de ventas de ese año n en todas las categorías de producto. Es decir, desea mantener el filtro en el año pero quitarlo en ProductCategoryName cuando calcule el denominador del porcentaje.

La tabla siguiente muestra los resultados cuando se crea nueva medida, llamada Reseller Sales CategoryName, usando la fórmula de la sección de códigos. Para ver cómo funciona esto, agregue el campo CalendarYear al área **Etiquetas de fila** de la tabla dinámica, y agregue el campo ProductCategoryName al área **Etiquetas de columna**. A continuación, agregue la nueva medida al área **Valores** de la tabla dinámica. Para ver los resultados como porcentajes, utilice las características de formato de Excel para aplicar un formato de número de porcentaje a las celdas que contengan la nueva medida, Reseller Sales CategoryName.

Reseller Sales CategoryName	Etiquetas de columna				
Etiquetas de fila	Accessories	Bikes	Clothing	Components	Grand Total
2005	0.25%	91.76%	0.42%	7.57%	100.00%
2006	0.38%	82.64%	1.99%	14.99%	100.00%
2007	0.90%	79.42%	2.67%	17.01%	100.00%
2008	0.99%	83.69%	2.37%	12.96%	100.00%
Total general	0.70%	82.47%	2.18%	14.65%	100.00%

## Código

```
=SUMX(ResellerSales_USD, ResellerSales_USD[SalesAmount_USD])/CALCULATE(SUM(ResellerSales_USD[SalesAmount_USD]), ALL(ProductCategory[ProductCategoryName]))
```

## Comentarios

La fórmula se construye del siguiente modo:

1. El numerador, `SUMX(ResellerSales_USD, ResellerSales_USD[SalesAmount_USD])`, es la suma de los valores de ResellerSales\_USD[SalesAmount\_USD] para la celda actual de la tabla dinámica, con los filtros de contexto aplicados en los campos CalendarYear y ProductCategoryName.
2. Para el denominador, use la función All(Column) para quitar el filtro en ProductCategoryName y calcule la suma de las filas restantes en la tabla ResellerSales\_USD, después de aplicar los filtros de contexto existentes desde las etiquetas de fila. El efecto neto es que para el denominador, la suma se calcula en el valor Year seleccionado (el filtro de contexto implícito) y para todos los valores de ProductCategoryName.

Para obtener más información acerca de la creación de medidas, vea [Create a Measure](#).

### Nota

Este ejemplo usa las tablas ResellerSales\_USD, DateTime y ProductCategory del libro de ejemplo de DAX. Para obtener más información acerca de los datos de ejemplo, vea [Obtener datos de ejemplo](#).

# Función ALLEXCEPT

Quita todos los filtros de contexto de la tabla excepto los filtros que se han aplicado a las columnas especificadas.

## Sintaxis

ALLEXCEPT (<table>,<column>[,<column>[,...]])

## Parámetros

Término	Definición
table	La tabla de la que se quitan todos los filtros de contexto, excepto los filtros de la columnas que se especifican en los argumentos subsiguientes.
column	La columna en la que se deben mantener los filtros de contexto.

El primer argumento para la función ALLEXCEPT debe ser una referencia a una tabla base; todos los argumentos subsiguientes deben ser referencias a las columnas base. Con la función ALLEXCEPT no se pueden usar expresiones de tabla o de columna.

## Valor devuelto

Tabla con todos los filtros quitados excepto los filtros de las columnas especificadas.

## Comentarios

Esta función no se usa por sí sola, sino que sirve de función intermedia que se puede usar para cambiar el conjunto de resultados en los que se realizan otros cálculos.

Tal y como se describe en la siguiente tabla, puede utilizar funciones ALL y ALLEXCEPT en escenarios diferentes.

Función y uso	Descripción
ALL(Table)	Quita todos los filtros de la tabla especificada. La función ALL(Table) devuelve todos los valores de la tabla y quita los filtros del contexto que, de otra forma, se podrían haber aplicado. Esta función resulta útil cuando se trabaja con muchos niveles de agrupación y se desea elaborar un cálculo que cree una proporción de un valor agregado al valor total.
ALL (Column[, Column[, ...]])	Quita todos los filtros de las columnas especificadas de la tabla; todos los demás filtros de las demás columnas de la tabla se siguen aplicando. Todos los argumentos de columna deben proceder de la misma tabla. La variante ALL(Column) resulta útil cuando se desea quitar los filtros de contexto de una o varias columnas específicas, y conservar todos los demás filtros de contexto.
ALLEXCEPT(Table, Column1 [,Column2]...)	Quita todos los filtros de contexto de la tabla excepto los filtros que se aplican a las columnas especificadas. Es un acceso directo conveniente para las situaciones en las que desea quitar los filtros en muchas columnas de una tabla, pero no en todas.

## Ejemplo

En el siguiente ejemplo se presenta una fórmula que puede utilizar en una medida. Para obtener más información acerca de cómo crear una medida, vea [Create a Measure](#).

La fórmula suma SalesAmount\_USD y usa la función ALLEXCEPT para quitar los centros de contexto en la tabla DateTime excepto si el filtro se ha aplicado a la columna CalendarYear.

#### Nota

En el ejemplo anterior se usan las tablas ResellerSales\_USD y DateTime del libro de ejemplo de DAX. Para obtener más información acerca de los datos de ejemplo, vea [Obtener datos de ejemplo](#).

```
=CALCULATE (SUM (ResellerSales_USD [SalesAmount_USD]), ALLEXCEPT (DateTime,  
DateTime [CalendarYear]))
```

Como la fórmula usa ALLEXCEPT, siempre que se use cualquier columna de la tabla DateTime, excepto CalendarYear, para segmentar la tabla dinámica, la fórmula quitará los filtros de segmentaciones, proporcionando un valor igual a la suma de SalesAmount\_USD para el valor de etiqueta de columna, tal como se muestra en la tabla 1.

No obstante, si se usa la columna CalendarYear para segmentar la tabla dinámica, los resultados son distintos. Como CalendarYear se ha especificado como el argumento para ALLEXCEPT, cuando los datos se segmentan en el año, se aplicará un filtro en los años en el nivel de fila, tal como se muestra en la tabla 2. Se recomienda al usuario que compare estas tablas para entender el comportamiento de ALLEXCEPT().

# Función ALLNOBLANKROW

A partir de la tabla primaria de una relación, devuelve todas las filas menos la fila en blanco, o todos los valores distintos de una columna menos la fila en blanco, y omite cualquier filtro de contexto que pudiera existir.

## Sintaxis

ALLNOBLANKROW (<table>|<column>)

## Parámetros

Término	Definición
table	Tabla de la que se quitan todos los filtros de contexto.
columna	Columna de la que se quitan todos los filtros de contexto.

Se debe pasar solo un parámetro; el parámetro es una tabla o una columna.

## Valor devuelto

Una tabla, cuando el parámetro que se pasó era una tabla, o una columna de valores, cuando el parámetro que se pasó era una columna.

## Comentarios

La función ALLNOBLANKROW solo filtra la fila en blanco que una tabla primaria, de una relación, mostrará cuando haya una o varias filas en la tabla secundaria que tengan valores no coincidentes con la columna primaria. Vea el ejemplo siguiente para obtener una explicación detallada.

En la siguiente tabla se resumen las variaciones de TODOS que se proporcionan en DAX, y sus diferencias:

Función y uso	Descripción
ALL(Column)	Quita todos los filtros de la columna especificada de la tabla; todos los demás filtros en las otras columnas de la tabla se siguen aplicando.
ALL(Table)	Quita todos los filtros de la tabla especificada.
ALLEXCEPT(Table,Col1,Col2...)	Invalida todos los filtros de contexto en la tabla excepto los de las columnas especificadas.
ALLNOBLANK(table column)	A partir de la tabla primaria de una relación, devuelve todas las filas menos la fila en blanco, o todos los valores distintos de una columna menos la fila en blanco, y omite cualquier filtro de contexto que pudiera existir.

Para obtener una descripción general del funcionamiento de ALL, junto con ejemplos paso a paso que usan ALL(Table) y ALL(Column), vea [Función ALL](#).

## Ejemplo

En los datos de ejemplo, la tabla ResellerSales\_USD contiene una fila que no tiene ningún valor y por consiguiente no se puede relacionar con ninguna de las tablas primarias en las relaciones dentro del libro. Utilizará esta tabla en una tabla dinámica para poder ver el comportamiento de la fila en blanco y cómo tratar los contadores en los datos no relacionados.

### Paso 1: comprobar los datos no relacionados

Abra la **Ventana de PowerPivot** y a continuación seleccione la tabla ResellerSales\_USD. En la columna ProductKey, filtre según los valores en blanco. Una fila permanecerá. En esa fila, todos los valores de las columnas deberían estar en blanco excepto SalesOrderLineNumber.

## Paso 2: crear una tabla dinámica

Cree una nueva tabla dinámica y, a continuación, arrastre la columna, `datetime.[Calendar Year]`, al panel Etiquetas de fila. En la tabla siguiente se muestran los resultados esperados:

Etiquetas de fila
2005
2006
2007
2008
Grand Total

Tenga en cuenta la etiqueta en blanco entre **2008** y **Grand Total**. Esta etiqueta en blanco representa el miembro desconocido, que es un grupo especial que se crea para tener en cuenta los valores de la tabla secundaria que no tengan ningún valor coincidente en la tabla primaria: en este ejemplo, la columna `datetime.[Calendar Year]`.

Cuando ve esta etiqueta en blanco en la tabla dinámica, sabe que en algunas de las tablas que se relacionan con la columna, `datetime.[Calendar Year]`, hay valores en blanco o valores no coincidentes. La tabla primaria es la que muestra la etiqueta en blanco, pero las filas que no coinciden están en una o en varias de las tablas secundarias. Las filas que se agregan a este grupo de etiquetas en blanco son alguno de los valores que no coinciden con ningún valor de la tabla primaria (por ejemplo, una fecha que no existe en la tabla de fecha y hora) o valores nulos, lo que significa que no hay ningún valor para la fecha. En este ejemplo, hemos colocado un valor en blanco en todas las columnas de la tabla de ventas secundaria. El que haya más valores en la tabla primaria que en las tablas de secundarias no supone ningún problema.

## Paso 3: contar filas con ALL y ALLNONBLANK

Agregue las siguientes dos medidas a la tabla de fecha y hora, para contar las filas de la tabla: **Countrows ALLNONBLANK of datetime**, **Countrows ALL of datetime**. Las fórmulas que puede usar para definir estas medidas se proporcionan en la siguiente sección de códigos.

En una tabla dinámica en blanco, agregue la columna `datetime.[Calendar Year]` a las etiquetas de fila y, a continuación, agregue las medidas recién creadas. Los resultados deberían ser similares a los de la tabla siguiente:

Etiquetas de fila	Countrows ALLNONBLANK of datetime	Countrows ALL of datetime
2005	1280	1281
2006	1280	1281
2007	1280	1281
2008	1280	1281
	1280	1281
Total general	1280	1281

Los resultados muestran una diferencia de una fila en el recuento de filas de la tabla. Sin embargo, si abre la **Ventana de PowerPivot** selecciona la tabla `datetime`, no puede encontrar ninguna fila en blanco en la tabla porque la fila en blanco especial mencionada aquí es el miembro desconocido.

## Paso 4: comprobar que el recuento es preciso



Para demostrar que ALLNOBLANKROW no cuenta ninguna fila realmente en blanco y solo trata la fila en blanco especial de la tabla primaria, agregue las siguientes dos medidas a la tabla ResellerSales\_USD: **Countrows ALLNOBLANKROW of ResellerSales\_USD,Countrows ALL of ResellerSales\_USD**. Cree una nueva tabla dinámica y arrastre la columna, datetime.[Calendar Year], al panel Etiquetas de fila. Ahora agregue las medidas recién creadas. Los resultados deberían ser similares a los siguientes:

Etiquetas de fila	Countrows ALLNOBLANKROW of ResellerSales_USD	Countrows ALL of ResellerSales_USD
2005	60856	60856
2006	60856	60856
2007	60856	60856
2008	60856	60856
	60856	60856
Total general	60856	60856

Ahora las dos medidas tienen los mismos resultados. Eso se debe a que la función ALLNOBLANKROW no cuenta las filas en blanco verdaderas de una tabla, sino que solo trata la fila en blanco que es un caso especial generado en una tabla primaria, cuando una o varias de las tablas secundarias de la relación contienen valores no coincidentes o valores en blanco.

```
// Countrows ALLNOBLANK of datetime
= COUNTROWS (ALLNOBLANKROW ('DateTime'))

// Countrows ALL of datetime
= COUNTROWS (ALL ('DateTime'))

// Countrows ALLNOBLANKROW of ResellerSales_USD
=COUNTROWS (ALLNOBLANKROW ('ResellerSales_USD'))

// Countrows ALL of ResellerSales_USD
=COUNTROWS (ALL ('ResellerSales_USD'))
```

# Función ALLSELECTED (DAX)

Quita los filtros de contexto de las columnas y filas de la consulta actual, a la vez que mantiene todos los demás filtros de contexto o filtros explícitos.

La función ALLSELECTED obtiene el contexto que representa todas las filas y columnas de la consulta, a la vez que conserva los filtros explícitos y los contextos que no sean filtros de fila y de columna. Esta función se puede utilizar para obtener totales visuales en las consultas.

## Sintaxis

```
ALLSELECTED([<tableName> | <columnName>])
```

## Parámetros

Nombre de una tabla existente, con la sintaxis estándar de DAX. Este parámetro no puede ser una expresión. Este parámetro es opcional.

Nombre, normalmente completo, de una columna existente que utiliza sintaxis estándar de DAX. No puede ser una expresión. Este parámetro es opcional.

## Valor devuelto

Contexto de la consulta, sin ningún filtro de columnas y filas.

## Comentarios

- Esta función tiene un argumento o ninguno. Si hay un argumento, el argumento es *tableName* o *columnName*.
- Esta función no es igual que ALL(), porque mantiene todos los filtros establecidos explícitamente en la consulta, y conserva todos los filtros de contexto excepto los filtros de fila y de columna.

## Ejemplo

En el ejemplo siguiente se muestra cómo generar diferentes niveles de totales visuales en un informe de tabla mediante expresiones DAX. En el informe, se han aplicado los dos (2) filtros anteriores a los datos de ventas de distribuidor: uno a Sales Territory Group = *Europe* y el otro a Promotion Type = *Volume Discount*. Una vez aplicados los filtros, se pueden calcular los totales visuales para todo el informe, para todos los años o para todas las categorías de producto. Además, con fines meramente ilustrativos, se obtiene también el total general de todas las ventas, quitando todos los filtros del informe. La evaluación de la expresión DAX siguiente da como resultado una tabla con toda la información necesaria para generar una tabla con totales visuales.

```
define
measure 'Reseller Sales'[Reseller Sales Amount]=sum('Reseller Sales'[Sales Amount])
measure 'Reseller Sales'[Reseller Grand Total]=calculate(sum('Reseller Sales'[Sales Amount]), ALL('Reseller Sales'))
measure 'Reseller Sales'[Reseller Visual Total]=calculate(sum('Reseller Sales'[Sales Amount]), ALLSELECTED())
measure 'Reseller Sales'[Reseller Visual Total for All of Calendar Year]=calculate(sum('Reseller Sales'[Sales Amount]), ALLSELECTED('Date'[Calendar Year]))
measure 'Reseller Sales'[Reseller Visual Total for All of Product Category Name]=calculate(sum('Reseller Sales'[Sales Amount]), ALLSELECTED('Product Category'[Product Category Name]))
evaluate
CalculateTable(
    //CT table expression
```

```

summarize(
//summarize table expression
crossjoin(distinct('Product Category'[Product Category Name]), distinct('Date'[Calendar
Year])))
//First Group by expression
, 'Product Category'[Product Category Name]
//Second Group by expression
, 'Date'[Calendar Year]
//Summary expressions
, "Reseller Sales Amount", [Reseller Sales Amount]
, "Reseller Grand Total", [Reseller Grand Total]
, "Reseller Visual Total", [Reseller Visual Total]
, "Reseller Visual Total for All of Calendar Year", [Reseller Visual Total for All of
Calendar Year]
, "Reseller Visual Total for All of Product Category Name", [Reseller Visual Total for
All of Product Category Name]
)
//CT filters
, 'Sales Territory'[Sales Territory Group]="Europe", 'Promotion'[Promotion
Type]="Volume Discount"
)
order by [Product Category Name], [Calendar Year]

```

Tras ejecutar la expresión anterior en SQL Server Management Studio para AdventureWorks DW Tabular Model 2012 se obtienen los resultados siguientes:

<b>[Product Category Name]</b>	<b>[Calendar Year]</b>	<b>[Reseller Sales Amount]</b>	<b>[Reseller Grand Total]</b>	<b>[Reseller Visual Total]</b>	<b>[Reseller Visual Total for All of Calendar Year]</b>	<b>[Reseller Visual Total for All of Product Category Name]</b>
Accessories	2000		80450596.9823	877006.7987	38786.018	
Accessories	2001		80450596.9823	877006.7987	38786.018	
Accessories	2002	625.7933	80450596.9823	877006.7987	38786.018	91495.3104
Accessories	2003	26037.3132	80450596.9823	877006.7987	38786.018	572927.0136
Accessories	2004	12122.9115	80450596.9823	877006.7987	38786.018	212584.4747
Accessories	2005		80450596.9823	877006.7987	38786.018	
Accessories	2006		80450596.9823	877006.7987	38786.018	
Bikes	2000		80450596.9823	877006.7987	689287.7939	
Bikes	2001		80450596.9823	877006.7987	689287.7939	

Bikes	2002	73778.938	80450596.9823	877006.7987	689287.7939	91495.3104
Bikes	2003	439771.4136	80450596.9823	877006.7987	689287.7939	572927.0136
Bikes	2004	175737.4423	80450596.9823	877006.7987	689287.7939	212584.4747
Bikes	2005		80450596.9823	877006.7987	689287.7939	
Bikes	2006		80450596.9823	877006.7987	689287.7939	
Clothing	2000		80450596.9823	877006.7987	95090.7757	
Clothing	2001		80450596.9823	877006.7987	95090.7757	
Clothing	2002	12132.4334	80450596.9823	877006.7987	95090.7757	91495.3104
Clothing	2003	58234.2214	80450596.9823	877006.7987	95090.7757	572927.0136
Clothing	2004	24724.1209	80450596.9823	877006.7987	95090.7757	212584.4747
Clothing	2005		80450596.9823	877006.7987	95090.7757	
Clothing	2006		80450596.9823	877006.7987	95090.7757	
Components	2000		80450596.9823	877006.7987	53842.2111	
Components	2001		80450596.9823	877006.7987	53842.2111	
Components	2002	4958.1457	80450596.9823	877006.7987	53842.2111	91495.3104
Components	2003	48884.0654	80450596.9823	877006.7987	53842.2111	572927.0136
Components	2004		80450596.9823	877006.7987	53842.2111	212584.4747
Components	2005		80450596.9823	877006.7987	53842.2111	
Components	2006		80450596.9823	877006.7987	53842.2111	

Las columnas del informe son:

El valor real de las ventas de revendedor por año y categoría de producto. Este valor aparece en una celda en el centro del informe, en la intersección de año y categoría.

El valor total de una categoría de producto para todos los años. Este valor aparece al final de una columna o fila de una categoría de producto determinada y en todos los años en el informe.

El valor total de un año en todas las categorías de producto. Este valor aparece al final de una columna o fila de un año determinado y en todas las categorías de producto en el informe.

El valor total de todos los años y categorías de producto. Este valor aparece normalmente en la esquina inferior derecha de la tabla.

Este es el total de todas las ventas de distribuidor antes de que se haya aplicado un filtro; observe la diferencia con [Reseller Visual Total]. Recuerde que este informe incluye dos (2) filtros, uno aplicado a Product Category Group y el otro aplicado a Promotion Type.

En el ejemplo siguiente se muestra el uso de ALLSELECTED() sin argumentos, para mostrar cómo calcular una proporción sobre el valor total mostrado en una tabla que se ha filtrado con segmentaciones de datos horizontales y

verticales. En este ejemplo se usa SQL Server 2012, PowerPivot para Excel y [Datos de ejemplo de PowerPivot \(DAX AdventureWorks\)](#).

- En la lista de campos de PowerPivot, arrastre la columna, ResellerSales[SalesAmount\_USD] al área **Valores**.
- Arrastre Promotion[PromotionType] al área **Segmentaciones de datos verticales**. Seleccione las segmentaciones de datos para Discontinued Product, Excess Inventory y Seasonal Discount.
- Arrastre SalesTerritory[SalesTerritoryGroup] al área **Segmentaciones de datos horizontales**. Seleccione la segmentación de datos de Europe.
- El valor de **Sum Of SalesAmount\_USD** debe ser 19.614,37 dólares.
- Arrastre ProductCategory[ProductCategoryName] al área **Etiquetas de fila** y DateTime[CalendarYear] al área **Etiquetas de columna**.

La tabla debe tener un aspecto parecido a:

	<b>SalesTerritoryGroup</b>					
	<i>Europe</i>	North America				
	Pacific	N/D				
<b>PromotionType</b>						
<i>Discontinued Product</i>		<b>Sum of SalesAmount_USD</b>	<b>Etiquetas de columna</b>			
<i>Excess Inventory</i>		<b>Etiquetas de fila</b>	<b>2006</b>	<b>2007</b>	<b>2008</b>	<b>Gr To</b>
New Product		Accessories	\$1,111.22	\$3,414.43		<b>\$4</b>
No Discount		Bikes	\$8,834.94		\$6,253.78	<b>\$1</b>
<i>Seasonal Discount</i>		<b>Grand Total</b>	<b>\$9,946.16</b>	<b>\$3,414.43</b>	<b>\$6,253.78</b>	<b>\$1</b>
Volume Discount						

- Use una calculadora para comprobar que la cantidad 3.414,43 de dólares (accessories en 2007) es el 17,41% de 19.614,37 dólares.
- Use una calculadora para comprobar que la cantidad 6.253,78 dólares (bicicletas en 2008) es el 31,88% de 19.614,37 dólares.
- Use una calculadora para comprobar que la cantidad 15.088,72 dólares (total general de las bicicletas) es el 76,93% de 19.614,37 dólares.
- Quite ResellerSales[SalesAmount\_USD] del área **Valores**.
- Cree una medida denominada **Reseller Sales Ratio** en la tabla ResellerSales, con la siguiente fórmula (dé formato de porcentaje al resultado):

=SUM(ResellerSales\_USD[SalesAmount\_USD]) /

CALCULATE(SUM(ResellerSales\_USD[SalesAmount\_USD]), ALLSELECTED())

La tabla debe tener un aspecto parecido al siguiente:

	<b>SalesTerritoryGroup</b>					
--	----------------------------	--	--	--	--	--

	<i>Europe</i>	North America				
	Pacific	N/D				
<b>PromotionType</b>						
<i>Discontinued Product</i>		<b>Reseller Sales Ratio</b>	<b>Etiquetas de columna</b>			
<i>Excess Inventory</i>		<b>Etiquetas de fila</b>	<b>2006</b>	<b>2007</b>	<b>2008</b>	<b>Grand Total</b>
New Product		Accessories	5.67 %	17.41 %		<b>23.07 %</b>
No Discount		Bikes	45.04 %		31.88 %	<b>76.93 %</b>
<i>Seasonal Discount</i>		<b>Grand Total</b>	<b>50.71 %</b>	<b>17.41 %</b>	<b>31.88 %</b>	<b>100.00 %</b>
Volume Discount						

- Compare los resultados de la calculadora con los de **Reseller Sales Ratio**, que deben coincidir. El valor del denominador está fijo en el valor de la parte inferior de la tabla.
- Como en la fórmula CALCULATE del denominador se utiliza la función ALLSELECTED, el denominador representa el total general de ventas, en la parte inferior de la tabla, una vez que se han aplicado las segmentaciones de datos verticales, horizontales y de página, pero antes de que se apliquen las segmentaciones de datos de fila y de columna.

**Nota:** si tiene filtros explícitos en la expresión, los filtros también se aplican a la expresión.

# Función CALCULATE

SQL Server 2014

[Otras versiones](#)



Evalúa una expresión en un contexto que está modificado por los filtros especificados.

## Sintaxis

CALCULATE (<expression>, <filter1>, <filter2>...)

## Parámetros

Término	Definición
expression	Expresión que se va a evaluar.
filter1, filter2, ...	(opcional) Lista separada por comas de expresiones booleanas o una expresión de tabla que define un filtro.

La expresión usada como el primer parámetro es, esencialmente, la misma que una medida.

Las siguientes restricciones se aplican a las expresiones booleanas que se usan como argumentos:

- La expresión no puede hacer referencia a una medida.
- La expresión no puede usar una función CALCULATE anidada.
- La expresión no puede usar ninguna función que recorra una tabla o devuelva una tabla, incluidas las funciones de agregación.

Sin embargo, una expresión booleana puede usar cualquier función que busque un valor único o que calcule un valor escalar.

## Valor devuelto

El valor que es el resultado de la expresión.

## Comentarios

Si se han filtrado los datos, la función CALCULATE cambia el contexto en el que se filtran los datos y evalúa la expresión en el nuevo contexto que especifique. Por cada columna usada en un argumento de filtro, se quitan los filtros existentes en dicha columna y en su lugar se aplica el filtro usado en el argumento.

## Ejemplo

Para calcular la proporción de ventas actuales de los vendedores en relación con las ventas totales de los vendedores, agregue una medida a la tabla dinámica que calcule la suma de las ventas de la celda actual (el numerador) y la divida por las ventas totales (el denominador). Para asegurarse de que el denominador permanece igual independientemente del modo en el que la tabla dinámica filtre o agrupe los datos, la parte de la fórmula que representa el denominador debe usar una función ALL para borrar cualquier filtro y crear el total correcto.

La tabla siguiente muestra los resultados cuando se crea la nueva medida, llamada All Reseller Sales Ratio, usando la fórmula de la sección de códigos.

Para ver cómo funciona esto, agregue el campo CalendarYear al área **Etiquetas de fila** de la tabla dinámica, y agregue el campo ProductCategoryName al área **Etiquetas de columna**. A continuación, agregue la nueva medida al área **Valores** de la tabla dinámica. Para mostrar los números como porcentajes, aplique el formato de número de porcentaje al área de la tabla dinámica que contiene la nueva medida, All Reseller Sales Ratio.

All Reseller Sales	Etiquetas de columna				
--------------------	----------------------	--	--	--	--

Etiquetas de fila	Accessories	Bikes	Clothing	Components	Grand Total
2005	0.02%	9.10%	0.04%	0.75%	9.91%
2006	0.11%	24.71%	0.60%	4.48%	29.90%
2007	0.36%	31.71%	1.07%	6.79%	39.93%
2008	0.20%	16.95%	0.48%	2.63%	20.26%
Grand Total	0.70%	82.47%	2.18%	14.65%	100.00%

```
=( SUM('ResellerSales_USD'[SalesAmount_USD]))
/CALCULATE( SUM('ResellerSales_USD'[SalesAmount_USD])
,ALL('ResellerSales_USD'))
```

La expresión CALCULATE en el denominador permite que la expresión de suma incluya todas las filas en el cálculo. Esto invalida los filtros implícitos para CalendarYear y ProductCategoryName que existen para la parte del numerador de la expresión.



# Función CALCULATETABLE

Evalúa una expresión de tabla en un contexto modificado por los filtros dados.

## Sintaxis

CALCULATETABLE (<expression>, <filter1>, <filter2>, ...)

## Parámetros

Término	Definición
Expression	La expresión de tabla que se va a evaluar
filter1, filter2, ...	Expresión booleana o expresión de tabla que define un filtro

La expresión usada como primer parámetro debe ser una función que devuelva una tabla.

Las siguientes restricciones se aplican a las expresiones booleanas que se usan como argumentos:

- La expresión no puede hacer referencia a una medida.
- La expresión no puede usar una función CALCULATE anidada.
- La expresión no puede usar ninguna función que recorra una tabla o devuelva una tabla, incluidas las funciones de agregación.

Sin embargo, una expresión booleana puede usar cualquier función que busque un valor único o que calcule un valor escalar.

## Valor devuelto

Una tabla de valores.

## Comentarios

La función CALCULATETABLE cambia el contexto en el que se filtran los datos y evalúa la expresión en el nuevo contexto que especifique. Por cada columna usada en un argumento de filtro, se quitan los filtros existentes en dicha columna y en su lugar se aplica el filtro usado en el argumento.

Esta función es un sinónimo de la función RELATEDTABLE.

## Ejemplo

En el siguiente ejemplo se usa la función CALCULATETABLE para obtener la suma de Internet Sales para 2006. Este valor se usa después para calcular la proporción de Internet Sales en comparación con todas las ventas del año 2006. La siguiente tabla muestra los resultados a partir de la fórmula siguiente.

Etiquetas de fila	Internet SalesAmount_USD	CalculateTable 2006 Internet Sales	Internet Sales to 2006 ratio
2005	\$2,627,031.40	\$5,681,440.58	0.46
2006	\$5,681,440.58	\$5,681,440.58	1.00
2007	\$8,705,066.67	\$5,681,440.58	1.53
2008	\$9,041,288.80	\$5,681,440.58	1.59
Grand Total	\$26,054,827.45	\$5,681,440.58	4.59

```
=SUMX( CALCULATETABLE('InternetSales_USD', 'DateTime'[CalendarYear]=2006)  
      , [SalesAmount_USD])
```



# Función DISTINCT

Devuelve una tabla de una columna que contiene los valores distintos de la columna especificada. En otras palabras, los valores duplicados se quitan y solo se devuelven los valores únicos.

## Nota

Esta función no se puede usar para devolver valores en una sola columna de una hoja de cálculo; en vez de eso, se mira la función DISTICNT dentro de una fórmula para obtener una lista de los valores distintos que se pueden pasar a otra función y, a continuación, contarse, sumarse o usarse para otras operaciones.

## Sintaxis

DISTINCT (<column>)

## Parámetros

Término	Definición
column	Columna de la que se devuelven los valores únicos. O una expresión que devuelve una columna.

## Valor devuelto

Columna de valores únicos.

## Comentarios

Los resultados de DISTINCTestán afectados por el contexto de filtro actual. Por ejemplo, si usa la fórmula del siguiente ejemplo para crear una medida, los resultados cambiarán siempre que la tabla se haya filtrado para mostrar únicamente una determinada región o período de tiempo.

### Funciones relacionadas

La función VALUES es similar a DISTINCT; también se puede usar para devolver una lista de valores únicos y, por lo general, devolverá exactamente los mismos resultados que DISTINCT. No obstante, en algunos contextos VALUES devolverá un valor especial adicional.Para obtener más información, vea [Función VALUES](#).

## Ejemplo

La siguiente fórmula hueca el número de clientes únicos que han generado pedidos a través del canal de Internet La siguiente tabla ilustra los posibles resultados cuando la fórmula se agrega a una tabla dinámica.

=COUNTROWS (DISTINCT (InternetSales\_USD[CustomerKey]))

Tenga en cuenta que no puede pegar la lista de valores que devuelve DISTINCT directamente en una columna. En su lugar, debe pasar los resultados de la función DISTINCT a otra función que cuente, filtre o agregue valores mediante la lista. Para simplificar el ejemplo al máximo, a continuación se ofrece la tabla de los valores distintos que se han pasado a la función COUNTROWS.

Cientes de Internet únicos	Etiquetas de columna			
Etiquetas de fila	Accessories	Bikes	Clothing	Grand Total
2005		1013		1013
2006		2677		2677

2007	6792	4875	2867	9309
2008	9435	5451	4196	11377
Total general	15114	9132	6852	18484

Asimismo, observe que los resultados no son aditivos. Es decir, el número total de clientes únicos en 2007 no es la suma de clientes únicos de *Accessories*, *Bikes* y *Clothing* durante ese año. El motivo es que un cliente se puede contar en varios grupos.

# Función EARLIER

Devuelve el valor actual de la columna especificada en un paso de la evaluación exterior de la columna mencionada. EARLIER es útil para los cálculos anidados en que se desea usar un cierto valor como entrada para generar cálculos basados en esa entrada. En Microsoft Excel, estos cálculos solo se pueden realizar dentro del contexto de la fila actual; sin embargo, en DAX el valor de la entrada se puede almacenar y, a continuación, realizar el cálculo con los datos de la tabla completa. EARLIER se usa principalmente en el contexto de columnas calculadas.

## Sintaxis

EARLIER(<column>, <number>)

## Parámetros

Término	Definición
column	Columna o expresión que se resuelve en una columna.
num	(Opcional) Número positivo para el paso de la evaluación exterior. El siguiente nivel de evaluación exterior se representa mediante 1; dos niveles exteriores se representan mediante 2, etc. Cuando se omite, el valor predeterminado es 1.

## Valor de propiedad y valor devuelto

Valor actual de la fila, de la columna, en el número de pasos de la evaluación exterior.

## Excepciones

Descripción de los errores

## Comentarios

**EARLIER** es correcta si existe un contexto de fila anterior al principio del recorrido de tabla. En caso contrario, devuelve un error. El rendimiento de **EARLIER** puede ser lento porque, en teoría, tendrá que realizar un número de operaciones cercano al número total de filas (de la columna) multiplicado por el mismo número (en función de la sintaxis de la expresión). Por ejemplo, si hay 10 filas en la columna, se necesitarán aproximadamente unas 100 operaciones; si hay 100 filas, será necesario realizar unas 10.000 operaciones.

Nota
En la práctica, el motor analítico en memoria xVelocity (VertiPaq) realiza las optimizaciones para reducir el número real de cálculos, pero debería tener cuidado al crear fórmulas que impliquen recursividad.

## Ejemplo

Para describir el uso de EARLIER, es necesario generar una situación de ejemplo que calcule un valor de rango y, a continuación, usar ese valor de rango en otros cálculos. El ejemplo siguiente está basado en esta tabla simple, ProductSubcategory, que muestra las ventas totales para cada ProductSubcategory. A continuación se muestra la tabla final, que incluye la columna de clasificación.

ProductSubcategoryKey	EnglishProductSubcategoryName	TotalSubcategorySales	SubcategoryRanking
-----------------------	-------------------------------	-----------------------	--------------------

18	Bib-Shorts	\$156,167.88	18
26	Bike Racks	\$220,720.70	14
27	Bike Stands	\$35,628.69	30
28	Bottles and Cages	\$59,342.43	24
5	Bottom Brackets	\$48,643.47	27
6	Brakes	\$62,113.16	23
19	Caps	\$47,934.54	28
7	Chains	\$8,847.08	35
29	Cleaners	\$16,882.62	32
8	Cranksets	\$191,522.09	15
9	Derailleurs	\$64,965.33	22
30	Fenders	\$41,974.10	29
10	Forks	\$74,727.66	21
20	Gloves	\$228,353.58	12
4	Handlebars	\$163,257.06	17
11	Headsets	\$57,659.99	25
31	Helmets	\$451,192.31	9
32	Hydration Packs	\$96,893.78	20
21	Jerseys	\$699,429.78	7
33	Lights		36
34	Locks	\$15,059.47	33
1	Mountain Bikes	\$34,305,864.29	2
12	Mountain Frames	\$4,511,170.68	4
35	Panniers		36
13	Pedals	\$140,422.20	19
36	Pumps	\$12,695.18	34

2	Road Bikes	\$40,551,696.34	1
14	Road Frames	\$3,636,398.71	5
15	Saddles	\$52,526.47	26
22	Shorts	\$385,707.80	10
23	Socks	\$28,337.85	31
24	Tights	\$189,179.37	16
37	Tires and Tubes	\$224,832.81	13
3	Touring Bikes	\$13,334,864.18	3
16	Touring Frames	\$1,545,344.02	6
25	Vests	\$240,990.04	11
17	Wheels	\$648,240.04	8

## Crear un valor de rango

Una manera de obtener un valor de rango para un valor determinado en una fila consiste en contar el número de filas de la misma tabla que tienen un valor superior (o inferior) al valor que se está comparando. Esta técnica devuelve un valor en blanco o cero para el valor máximo de la tabla, mientras que los valores iguales tendrán el mismo valor de rango, y el valor siguiente (después de los valores iguales) tendrá un valor de rango no consecutivo. Vea el ejemplo siguiente.

Una nueva columna calculada, *SubCategorySalesRanking*, se crea mediante la fórmula siguiente.

```
= COUNTROWS (FILTER (ProductSubcategory,
EARLIER (ProductSubcategory[TotalSubcategorySales]) < ProductSubcategory[TotalSubcategorySales])) + 1
```

Los pasos siguientes describen el método de cálculo con más detalle.

1. La función **EARLIER** obtiene el valor de *TotalSubcategorySales* para la fila actual de la tabla. En este caso, puesto que el proceso se está iniciando, es la primera fila de la tabla
2. **EARLIER** ([*TotalSubcategorySales*]) se evalúa en 156.167,88 dólares, la fila actual del bucle exterior.
3. La función **FILTER** devuelve ahora una tabla en la que todas las filas tienen un valor de *TotalSubcategorySales* superior a 156.167,88 dólares (que es el valor actual de **EARLIER**).
4. La función **COUNTROWS** cuenta las filas de la tabla filtrada y asigna ese valor a la nueva columna calculada en la fila actual más 1. La suma de 1 es necesaria para evitar que el valor de rango superior se convierta en un valor en blanco.
5. La fórmula de la columna calculada se desplaza a la siguiente fila y repite los pasos 1 a 4. Estos pasos se repiten hasta que se llega al final de la tabla.

La función **EARLIER** siempre obtendrá el valor de la columna antes que la operación de tabla actual. Si necesita obtener un valor del bucle anterior, establezca el segundo argumento en 2.

# Función EARLIEST

Devuelve el valor actual de la columna especificada en un paso de la evaluación exterior de la columna especificada.

## Sintaxis

EARLIEST(<column>)

## Parámetros

Término	Definición
column	Una referencia a una columna.

## Valor de propiedad y valor devuelto

Una columna con los filtros quitados.

## Comentarios

La función EARLIEST es similar a EARLIER, pero le permite especificar un nivel adicional de recursividad.

## Ejemplo

Los datos de ejemplo actuales no admiten este escenario.

=EARLIEST(<column>)



# Función EARLIER

Devuelve el valor actual de la columna especificada en un paso de la evaluación exterior de la columna mencionada. EARLIER es útil para los cálculos anidados en que se desea usar un cierto valor como entrada para generar cálculos basados en esa entrada. En Microsoft Excel, estos cálculos solo se pueden realizar dentro del contexto de la fila actual; sin embargo, en DAX el valor de la entrada se puede almacenar y, a continuación, realizar el cálculo con los datos de la tabla completa. EARLIER se usa principalmente en el contexto de columnas calculadas.

## Sintaxis

EARLIER(<column>, <number>)

## Parámetros

Término	Definición
column	Columna o expresión que se resuelve en una columna.
num	(Opcional) Número positivo para el paso de la evaluación exterior. El siguiente nivel de evaluación exterior se representa mediante 1; dos niveles exteriores se representan mediante 2, etc. Cuando se omite, el valor predeterminado es 1.

## Valor de propiedad y valor devuelto

Valor actual de la fila, de la columna, en el número de pasos de la evaluación exterior.

## Excepciones

Descripción de los errores

## Comentarios

**EARLIER** es correcta si existe un contexto de fila anterior al principio del recorrido de tabla. En caso contrario, devuelve un error. El rendimiento de **EARLIER** puede ser lento porque, en teoría, tendrá que realizar un número de operaciones cercano al número total de filas (de la columna) multiplicado por el mismo número (en función de la sintaxis de la expresión). Por ejemplo, si hay 10 filas en la columna, se necesitarán aproximadamente unas 100 operaciones; si hay 100 filas, será necesario realizar unas 10.000 operaciones.

Nota
En la práctica, el motor analítico en memoria xVelocity (VertiPaq) realiza las optimizaciones para reducir el número real de cálculos, pero debería tener cuidado al crear fórmulas que impliquen recursividad.

## Ejemplo

Para describir el uso de EARLIER, es necesario generar una situación de ejemplo que calcule un valor de rango y, a continuación, usar ese valor de rango en otros cálculos. El ejemplo siguiente está basado en esta tabla simple, ProductSubcategory, que muestra las ventas totales para cada ProductSubcategory. A continuación se muestra la tabla final, que incluye la columna de clasificación.

ProductSubcategoryKey	EnglishProductSubcategoryName	TotalSubcategorySales	SubcategoryRanking
-----------------------	-------------------------------	-----------------------	--------------------

18	Bib-Shorts	\$156,167.88	18
26	Bike Racks	\$220,720.70	14
27	Bike Stands	\$35,628.69	30
28	Bottles and Cages	\$59,342.43	24
5	Bottom Brackets	\$48,643.47	27
6	Brakes	\$62,113.16	23
19	Caps	\$47,934.54	28
7	Chains	\$8,847.08	35
29	Cleaners	\$16,882.62	32
8	Cranksets	\$191,522.09	15
9	Derailleurs	\$64,965.33	22
30	Fenders	\$41,974.10	29
10	Forks	\$74,727.66	21
20	Gloves	\$228,353.58	12
4	Handlebars	\$163,257.06	17
11	Headsets	\$57,659.99	25
31	Helmets	\$451,192.31	9
32	Hydration Packs	\$96,893.78	20
21	Jerseys	\$699,429.78	7
33	Lights		36
34	Locks	\$15,059.47	33
1	Mountain Bikes	\$34,305,864.29	2
12	Mountain Frames	\$4,511,170.68	4
35	Panniers		36
13	Pedals	\$140,422.20	19
36	Pumps	\$12,695.18	34

2	Road Bikes	\$40,551,696.34	1
14	Road Frames	\$3,636,398.71	5
15	Saddles	\$52,526.47	26
22	Shorts	\$385,707.80	10
23	Socks	\$28,337.85	31
24	Tights	\$189,179.37	16
37	Tires and Tubes	\$224,832.81	13
3	Touring Bikes	\$13,334,864.18	3
16	Touring Frames	\$1,545,344.02	6
25	Vests	\$240,990.04	11
17	Wheels	\$648,240.04	8

## Crear un valor de rango

Una manera de obtener un valor de rango para un valor determinado en una fila consiste en contar el número de filas de la misma tabla que tienen un valor superior (o inferior) al valor que se está comparando. Esta técnica devuelve un valor en blanco o cero para el valor máximo de la tabla, mientras que los valores iguales tendrán el mismo valor de rango, y el valor siguiente (después de los valores iguales) tendrá un valor de rango no consecutivo. Vea el ejemplo siguiente.

Una nueva columna calculada, *SubCategorySalesRanking*, se crea mediante la fórmula siguiente.

```
= COUNTROWS (FILTER (ProductSubcategory,
EARLIER (ProductSubcategory[TotalSubcategorySales]) < ProductSubcategory[TotalSubcategorySales])) + 1
```

Los pasos siguientes describen el método de cálculo con más detalle.

1. La función **EARLIER** obtiene el valor de *TotalSubcategorySales* para la fila actual de la tabla. En este caso, puesto que el proceso se está iniciando, es la primera fila de la tabla
2. **EARLIER** ([*TotalSubcategorySales*]) se evalúa en 156.167,88 dólares, la fila actual del bucle exterior.
3. La función **FILTER** devuelve ahora una tabla en la que todas las filas tienen un valor de *TotalSubcategorySales* superior a 156.167,88 dólares (que es el valor actual de **EARLIER**).
4. La función **COUNTROWS** cuenta las filas de la tabla filtrada y asigna ese valor a la nueva columna calculada en la fila actual más 1. La suma de 1 es necesaria para evitar que el valor de rango superior se convierta en un valor en blanco.
5. La fórmula de la columna calculada se desplaza a la siguiente fila y repite los pasos 1 a 4. Estos pasos se repiten hasta que se llega al final de la tabla.

La función **EARLIER** siempre obtendrá el valor de la columna antes que la operación de tabla actual. Si necesita obtener un valor del bucle anterior, establezca el segundo argumento en 2.

# Función FILTER

Devuelve una tabla que representa un subconjunto de otra tabla o expresión.

## Sintaxis

`FILTER(<table>,<filter>)`

## Parámetros

Término	Definición
table	Tabla que se va a filtrar. La tabla también puede ser una expresión que dé una tabla como resultado.
Filtro	Expresión booleana que se evaluará para cada fila de la tabla. Por ejemplo, <code>[Amount] &gt; 0</code> o <code>[Region] = "France"</code>

## Valor devuelto

Tabla que contiene solo las filas filtradas.

## Comentarios

Puede usar FILTER para reducir el número de filas en la tabla con la que está trabajando y usar solo datos específicos en los cálculos. FILTER no se usa independientemente, sino como una función integrada en otras funciones que requieren una tabla como argumento.

## Ejemplo

El ejemplo siguiente crea un informe de ventas en Internet fuera de los Estados Unidos usando una medida que filtra las ventas en el exterior y las clasifica por año y categoría de producto. Para crear esta medida, filtre la tabla, Internet Sales USD, usando Sales Territory, y, a continuación, emplee la tabla filtrada en una función SUMX.

En este ejemplo, la expresión `FILTER('InternetSales_USD', RELATED('SalesTerritory'[SalesTerritoryCountry])<>"United States")` devuelve una tabla que es un subconjunto de Internet Sales menos todas las filas que pertenecen al territorio de ventas de Estados Unidos. La función RELATED es lo que vincula la clave Territory en la tabla Internet Sales con SalesTerritoryCountry en la tabla SalesTerritory.

La siguiente tabla demuestra el concepto de la medida, NON USA Internet Sales, la fórmula que se proporciona a continuación en la sección de código. En la tabla se comparan todas las ventas en Internet con las ventas fuera de EE. UU. para mostrar que la expresión del filtro funciona, excluyendo del cálculo las ventas en Estados Unidos.

Para volver a crear esta tabla, agregue el campo SalesTerritoryCountry al área **Etiquetas de fila** de la tabla dinámica.

**Tabla 1. Comparar las ventas totales para EE. UU. con todas las demás regiones**

Etiquetas de fila	Internet Sales	Non USA Internet Sales
Australia	\$4,999,021.84	\$4,999,021.84
Canada	\$1,343,109.10	\$1,343,109.10
France	\$2,490,944.57	\$2,490,944.57
Germany	\$2,775,195.60	\$2,775,195.60
United Kingdom	\$5,057,076.55	\$5,057,076.55

United States	\$9,389,479.79	
Grand Total	\$26,054,827.45	\$16,665,347.67

La tabla de informe final muestra los resultados al crear una tabla dinámica usando la medida NON USA Internet Sales. Agregue el campo CalendarYear al área **Etiquetas de fila** de la tabla dinámica y agregue el campo ProductCategoryName al área **Etiquetas de columna**.

**Tabla 2.Comparar las ventas fuera de EE. UU.por categorías de producto**

Non USA Internet Sales	Etiquetas de columna			
Etiquetas de fila	Accessories	Bikes	Clothing	Grand Total
2005		\$1,526,481.95		\$1,526,481.95
2006		\$3,554,744.04		\$3,554,744.04
2007	\$156,480.18	\$5,640,106.05	\$70,142.77	\$5,866,729.00
2008	\$228,159.45	\$5,386,558.19	\$102,675.04	\$5,717,392.68
Total general	\$384,639.63	\$16,107,890.23	\$172,817.81	\$16,665,347.67

```
SUMX(FILTER('InternetSales_USD',
RELATED('SalesTerritory'[SalesTerritoryCountry])<>"United States")
,'InternetSales_USD'[SalesAmount_USD])
```

# Función FILTERS (DAX)

Devuelve los valores que se aplican directamente como filtros a *columnName*.

## Sintaxis

`FILTERS (<columnName>)`

## Parámetros

Nombre de una columna existente, con la sintaxis estándar de DAX. No puede ser una expresión.

## Valor devuelto

Valores que se aplican directamente como filtros a *columnName*.

## Ejemplo

En el ejemplo siguiente se muestra cómo determinar el número de filtros directos que tiene una columna.

```
=COUNTROWS (FILTERS (ResellerSales_USD[ProductKey]))
```

El ejemplo anterior permite saber cuántos filtros directos de ResellerSales\_USD[ProductKey] se han aplicado al contexto en que se evalúa la expresión.

# HASONEFILTER (Función) (DAX)

SQL Server 2014

[Otras versiones](#)



Devuelve **TRUE** cuando el número de valores filtrados directamente en *columnName* es uno; en caso contrario, devuelve **FALSE**.

## Sintaxis

HASONEFILTER(<columnName>)

## Parámetros

columnName

Nombre de una columna existente, con la sintaxis estándar de DAX. No puede ser una expresión.

## Valor devuelto

Devuelve **TRUE** cuando el número de valores filtrados directamente en *columnName* es uno; en caso contrario, devuelve **FALSE**.

## Comentarios

1. Esta función es parecida a HASONEVALUE(), con la diferencia de que HASONEVALUE() se basa en filtros cruzados, mientras que HASONEFILTER() funciona con un filtro directo.

## Ejemplo

En el ejemplo siguiente se muestra cómo utilizar HASONEFILTER() para obtener el filtro para ResellerSales\_USD [ProductKey] si hay un filtro, o para obtener BLANK si no hay filtros o hay más de un filtro para ResellerSales\_USD [ProductKey].

```
=IF (HASONEFILTER (ResellerSales_USD[ProductKey]), FILTERS (ResellerSales_USD[ProductKey]),  
BLANK ( ) )
```

# Función HASONEVALUE (DAX)

Devuelve **TRUE** cuando el contexto de *columnName* se ha filtrado hasta llegar a un único valor distintivo. En caso contrario, es **FALSE**.

## Sintaxis

HTML

HASONEVALUE (<columnName>)

## Parámetros

Nombre de una columna existente, con la sintaxis estándar de DAX. No puede ser una expresión.

## Valor devuelto

**TRUE** cuando el contexto de *columnName* se ha filtrado hasta llegar a un único valor distintivo. En caso contrario, es **FALSE**.

## Comentarios

- Una expresión equivalente a HASONEVALUE() es **COUNTROWS(VALUES(<columnName>)) = 1**.

## Ejemplo

En el siguiente ejemplo, se desea crear una fórmula que compruebe si el contexto se segmenta mediante un valor para calcular un porcentaje en un escenario predefinido. En este caso, se desea comparar las ventas de distribuidor con las ventas de 2007, por lo que necesita saber si el contexto se filtra por años individuales. Por otra parte, si la comparación carece de significado, desea que se devuelva BLANK.

Si desea seguir el escenario, puede descargar la hoja de cálculo con el modelo desde [Datos de ejemplo de PowerPivot](#). Cree una medida llamada [ResellerSales compared to 2007] mediante la siguiente expresión:

```
=IF (HASONEVALUE (DateTime [CalendarYear]) , SUM (ResellerSales_USD [SalesAmount_USD]) / CALCULATE (SUM (ResellerSales_USD [SalesAmount_USD]) , DateTime [CalendarYear]=2007) , BLANK ())
```

- Después de crear la medida, debe tener un resultado vacío en [ResellerSales compared to 2007], como se muestra a continuación:

ResellerSales compared to 2007

- La celda en blanco del resultado se debe a que no hay filtros de año único en ninguna parte del contexto.
- Arrastre DateTime[CalendarYear] al cuadro **Etiquetas de columna**. La tabla debería tener un aspecto parecido a este:

	Etiquetas de columna			
	2005	2006	2007	2008
ResellerSales compared to 2007	24.83 %	74.88 %	100.00 %	50.73 %

- Arrastre ProductCategory[ProductCategoryName] al cuadro **Etiquetas de fila** para obtener algo parecido a esto:

ResellerSales compared to 2007	Etiquetas de columna			
Etiquetas de fila	2005	2006	2007	2008
Accessories	6.74 %	31.40 %	100.00 %	55.58 %



Bikes	28.69 %	77.92 %	100.00 %	53.46 %
Clothing	3.90 %	55.86 %	100.00 %	44.92 %
Components	11.05 %	65.99 %	100.00 %	38.65 %
<b>Grand Total</b>	<b>24.83 %</b>	<b>74.88 %</b>	<b>100.00 %</b>	<b>50.73 %</b>

5. ¿Ha observado que los **totales generales** han aparecido en la parte inferior de las columnas, pero no para las filas? La razón es que el contexto para los totales generales de las filas comprende más de un año, pero para las columnas comprende un único año.
6. Arrastre DateTime[CalendarYear] al cuadro **Segmentaciones de datos horizontales** y SalesTerritory[SalesTerritoryGroup] al cuadro **Etiquetas horizontales**. Debería tener un conjunto de resultados vacío, porque la tabla contiene datos de varios años. Seleccione **2006** en la segmentación de datos; la tabla debería volver a tener datos. Pruebe otros años para ver cómo cambian los resultados.
7. En resumen, HASONEVALUE() permite identificar si la expresión se evalúa en el contexto de un solo valor para *columnName*.

# Función ISCROSSFILTERED (DAX)

Devuelve TRUE si se filtra *columnName*, u otra columna de la misma tabla o una tabla relacionada.

## Sintaxis

ISCROSSFILTERED (<columnName>)

## Parámetros

Nombre de una columna existente, con la sintaxis estándar de DAX. No puede ser una expresión.

## Valor devuelto

**TRUE** si se filtra *columnName*, u otra columna de la misma tabla o una tabla relacionada. En caso contrario, **FALSE**.

## Comentarios

- Se dice que columna está afectada por un filtro cruzado cuando un filtro aplicado a otra columna de la misma tabla o de una tabla relacionada afecta a *columnName* porque también la filtra. Se dice que una columna está filtrada *directamente* cuando se le aplican filtros.
- La función relacionada [Función ISFILTERED \(DAX\)](#) devuelve TRUE si *columnName* se filtra directamente.

## Ejemplo

En el ejemplo siguiente se crea una medida y, después, se presentan diferentes escenarios para explicar el comportamiento de ISCROSSFILTERED(). Los escenarios se pueden seguir descargando la hoja de cálculo [DAX AdventureWorks](#).

Primero, cree la medida siguiente [Is ProductName Cross Filtered] en la tabla [Product].

=ISCROSSFILTERED (Product [ProductName] )

Descripción de ISCROSSFILTERED():

- Después de crear la medida, la tabla dinámica debe mostrar que [Is ProductName Cross Filtered] es **FALSE**, porque la expresión no se está filtrando en absoluto. Ahora, debe tener algo parecido a:

Is ProductName Cross Filtered						
FALSE						

- Si no aparece nada en la tabla dinámica, arrastre la columna [Is ProductName Cross Filtered] al cuadro **Valores**.
- Arrastre DateTime[CalendarYear] al cuadro **Etiquetas de columna** y SalesTerritory[SalesTerritoryGroup] al cuadro **Etiquetas de fila**; todos los valores deben aparecer como **FALSE**. Ahora, debe tener algo parecido a:

Is Product Name CrossFiltered	Etiquetas de columna					
Etiquetas de fila	2005	2006	2007	2008		Grand T
Europe	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
N/D	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
North America	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
Pacific	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE

Total general	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE
---------------	-------	-------	-------	-------	-------	-------

4. Todas las celdas tienen el valor FALSE porque ni DateTime[CalendarYear] ni SalesTerritory[SalesTerritoryGroup] afectan a Product[ProductName].
5. Arrastre la columna ProductCategory[ProductCategoryName] al cuadro **Segmentaciones de datos horizontales** y ProductSubCategory[ProductSubCategoryName] al cuadro **Segmentaciones de datos verticales**. Todos los valores deben seguir apareciendo como **FALSE**, porque cuando agrega una columna al cuadro de segmentaciones de datos aún no ha seleccionado un conjunto de segmentación. Por consiguiente, debería tener algo parecido a esto:

	ProductCategoryName					
	<i>Accessories</i>	<i>Bikes</i>				
	<i>Clothing</i>	<i>Components</i>				
ProductSubcategoryName	Is Product Name CrossFiltered	Etiquetas de columna				
<i>Bib-Shorts</i>	Etiquetas de fila	2005	2006	2007	2008	
<i>Bike-Racks</i>	Europe	FALSE	FALSE	FALSE	FALSE	FALSE
<i>Bike-Stands</i>	N/D	FALSE	FALSE	FALSE	FALSE	FALSE
<i>Bottles and Cages</i>	North America	FALSE	FALSE	FALSE	FALSE	FALSE
<i>Bottom Brackets</i>	Pacific	FALSE	FALSE	FALSE	FALSE	FALSE
<i>Brakes</i>		FALSE	FALSE	FALSE	FALSE	FALSE
<i>Caps</i>	Total general	FALSE	FALSE	FALSE	FALSE	FALSE
<i>Chains</i>						
<i>Cleaners</i>						
<i>Cranksets</i>						
<i>Derailleurs</i>						
<i>Fenders</i>						
<i>Forks</i>						
<i>Gloves</i>						
<i>Handlebars</i>						
<i>Headsets</i>						

6. Seleccione cualquier artículo en las segmentaciones de datos y todas las celdas de la tabla cambiarán a TRUE, porque ahora se está filtrando la columna Product[ProductName] a través de las tablas relacionadas ProductCategory y ProductSubcategory. Los resultados deben tener un aspecto parecido al siguiente:

	ProductCategoryName						
	<i>Accessories</i>	<b><i>Bikes</i></b>					
	<i>Clothing</i>	<i>Components</i>					
ProductSubcategoryName	Is Product Name CrossFiltered	Etiquetas de columna					
<i>Bib-Shorts</i>	Etiquetas de fila	2005	2006	2007	2008		Gr To
<i>Bike-Racks</i>	Europe	TRUE	TRUE	TRUE	TRUE	TRUE	TF
<i>Bike-Stands</i>	N/D	TRUE	TRUE	TRUE	TRUE	TRUE	TF
<i>Bottles and Cages</i>	North America	TRUE	TRUE	TRUE	TRUE	TRUE	TF
<i>Bottom Brackets</i>	Pacific	TRUE	TRUE	TRUE	TRUE	TRUE	TF
<i>Brakes</i>		TRUE	TRUE	TRUE	TRUE	TRUE	TF
<i>Caps</i>	Total general	TRUE	TRUE	TRUE	TRUE	TRUE	TF
<i>Chains</i>							
<i>Cleaners</i>							
<i>Cranksets</i>							
<i>Derailleurs</i>							
<i>Fenders</i>							
<i>Forks</i>							
<i>Gloves</i>							
<i>Handlebars</i>							
<i>Headsets</i>							

7. Ahora puede reemplazar las segmentaciones de datos con Product[ProductName] y Product[ModelName]. Si no selecciona ningún artículo en ninguna segmentación de datos, la medida devuelve **FALSE**; sin embargo, si selecciona un artículo de segmentación de datos, la medida devuelve **TRUE** porque está filtrando Product[ProductName] mediante sí misma o mediante Product[ModelName].
8. En conclusión, ISCROSSFILTERED() le permite determinar si *columnName* se está filtrando en el contexto en que se evalúa la expresión de DAX.

En este ejemplo se usa el [modelo de la hoja de cálculo de ejemplo DAX AdventureWorks](#).

# Función ISFILTERED (DAX)

Devuelve TRUE si *columnName* se está filtrando directamente. Si no hay ningún filtro en la columna, o si el filtrado tiene lugar porque se está filtrando otra columna de la misma tabla o de tabla relacionada, la función devuelve **FALSE**.

## Sintaxis

ISFILTERED(<columnName>)

## Parámetros

Nombre de una columna existente, con la sintaxis estándar de DAX. No puede ser una expresión.

## Valor devuelto

TRUE si *columnName* se está filtrando directamente.

## Comentarios

- Se dice que *columnName* se filtra directamente cuando los filtros se aplican a la columna. Se dice que una columna está afectada por un filtro cruzado cuando un filtro aplicado a otra columna de la misma tabla o de una tabla relacionada afecta a la columna *columnName* y también la filtra.
- La función [Función ISCROSSFILTERED \(DAX\)](#) relacionada devuelve TRUE si se filtra *columnName* u otra columna de la misma tabla o una tabla relacionada.

## Ejemplo

En el ejemplo siguiente se crea una medida y, después, se muestran diferentes escenarios para explicar el comportamiento de ISFILTERED(). Los escenarios se pueden ver descargando la hoja de cálculo [Expresiones de análisis de datos \(DAX\)](#).

Primero, cree la medida siguiente [Is ProductName Filtered directly] en la tabla [Product].

=ISFILTERED(Product[ProductName])

Descripción de ISFILTERED():

- Después de crear la medida, la tabla dinámica debe mostrar que [Is ProductName Filtered directly] es FALSE, porque la expresión no se está filtrando en absoluto. Ahora, debe tener algo parecido a:

Is ProductName Filtered directly
FALSE

- Si no hay nada en la tabla dinámica, arrastre [Is ProductName Filtered directly] al cuadro **Values**.
- Arrastre ProductCategory[ProductCategoryName] al cuadro **Row Labels**; todos los valores deben seguir siendo **FALSE** y la tabla debe tener un aspecto parecido a este:

Etiquetas de fila	Is ProductName Filtered directly
Accessories	FALSE
Bikes	FALSE
Clothing	FALSE
Components	FALSE
	FALSE
Grand Total	FALSE

4. El motivo por el que la tabla contiene valores FALSE es que ProductCategory[ProductCategoryName] filtra de forma indirecta o cruzada Product[ProductName], pero no es un filtro directo de la columna.
5. Arrastre Product[Status] al cuadro **Etiquetas de columna**; todos los valores deben seguir siendo **FALSE** y la tabla debe tener un aspecto parecido a este:

Is ProductName Filtered directly	Etiquetas de columna		
Etiquetas de fila		Current	Grand Total
Accessories	FALSE	FALSE	FALSE
Bikes	FALSE	FALSE	FALSE
Clothing	FALSE	FALSE	FALSE
Components	FALSE	FALSE	FALSE
	FALSE	FALSE	FALSE
<b>Grand Total</b>	<b>FALSE</b>	<b>FALSE</b>	<b>FALSE</b>

6. La tabla tiene todavía muchos valores **FALSE**, porque Product[Status] no filtra Product[Name] directamente.
7. El último paso de este proceso es arrastrar Product[Name] al cuadro **Report Filter**. En este momento, no debe haber cambiado nada; sin embargo, una vez que haya seleccionado un nombre de producto, los valores de la tabla deben cambiar a **TRUE**. Dependiendo del producto que haya seleccionado, la tabla debe tener un aspecto parecido a este:

ProductName	Mountain-400-W Silver, 38	
Is ProductName Filtered directly	Etiquetas de columna	
Etiquetas de fila	Current	Grand Total
Accessories	TRUE	TRUE
Bikes	TRUE	TRUE
Clothing	TRUE	TRUE
Components	TRUE	TRUE
	TRUE	TRUE
<b>Grand Total</b>	<b>TRUE</b>	<b>TRUE</b>

8. En conclusión, ISFILTERED() le permite determinar si columnName se está filtrando directamente en el contexto en que se evalúa la expresión.

En este ejemplo se usa el [modelo de la hoja de cálculo de ejemplo DAX AdventureWorks](#).

# Función RELATED

SQL Server 2014

[Otras versiones](#)

Devuelve un valor relacionado de otra tabla.

## Sintaxis

RELATED (<column>)

## Parámetros

Término	Definición
column	Columna que contiene los valores que desea recuperar.

## Valor devuelto

Valor único que se relaciona con la fila actual.

## Comentarios

La función RELATED requiere la existencia de una relación entre la tabla actual y la tabla con información relacionada. Debe especificar la columna que contiene los datos que desea y la función sigue una relación existente de varios a uno para capturar el valor de la columna especificada en la tabla relacionada.

Si no existe una relación, debe crear una. Para obtener más información, vea [Creating a Relationship](#).

Cuando la función RELATED realiza una búsqueda, examina todos los valores en la tabla especificada independientemente de los filtros que se puedan haber aplicado.

### Nota

La función RELATED necesita un contexto de fila. Por consiguiente, solo se puede usar en una expresión de columna calculada, donde el contexto de la fila actual es inequívoco, o como una función anidada en una expresión que usa una función de examen de tablas. Una función de recorrido de tabla, como SUMX, obtiene el valor a partir del valor de la fila actual y, a continuación, examina otra tabla para encontrar instancias de ese valor.

## Ejemplo

En el siguiente ejemplo, se crea la medida Non USA Internet Sales para generar un informe de ventas que excluye las ventas en los Estados Unidos. Para crear la medida, la tabla InternetSales\_USD se debe filtrarse para excluir todas las ventas que pertenecen a los Estados Unidos en la tabla SalesTerritory. Los Estados Unidos, como país, aparece cinco veces en la tabla SalesTerritory; una vez para cada una de las siguientes regiones: Northwest, Northeast, Central, Southwest y Southeast.

El primer enfoque para filtrar Internet Sales para crear la medida podría ser crear una expresión de filtro como la siguiente:

```
FILTER('InternetSales_USD', 'InternetSales_USD'[SalesTerritoryKey]<>1 &&  
'InternetSales_USD'[SalesTerritoryKey]<>2 &&  
'InternetSales_USD'[SalesTerritoryKey]<>3 &&  
'InternetSales_USD'[SalesTerritoryKey]<>4 &&  
'InternetSales_USD'[SalesTerritoryKey]<>5)
```

Sin embargo, este enfoque es susceptible a errores tipográficos y podría no funcionar si alguna de las regiones existentes se divide en el futuro.



Una solución más apropiada sería usar la relación existente entre InternetSales\_USD y SalesTerritory, e indicar explícitamente que el país debe ser diferente de Estados Unidos. Para esto, cree una expresión de filtro como la siguiente:

```
FILTER( 'InternetSales_USD',  
RELATED('SalesTerritory'[SalesTerritoryCountry])<>"United States")
```

Esta expresión utiliza la función RELATED para buscar el valor de país en la tabla SalesTerritory, comenzando con el valor de la columna de clave, SalesTerritoryKey, en la tabla InternetSales\_USD. La función de filtro usa el resultado de la búsqueda para determinar si se filtra la fila InternetSales\_USD o no.

Nota
Si el ejemplo no funciona, es posible que deba crear una relación entre las tablas. Para obtener más información, vea <a href="#">Relationships Between Tables</a> .

```
= SUMX ( FILTER ( 'InternetSales_USD'  
    , RELATED ( 'SalesTerritory' [SalesTerritoryCountry]  
        <> "United States"  
    )  
    , 'InternetSales_USD' [SalesAmount_USD] )
```

La siguiente tabla solo muestra totales para cada región, para demostrar que la expresión de filtro en la medida, No USA Internet Sales, funciona tal y como se tenía previsto.

Etiquetas de fila	Internet Sales	Non USA Internet Sales
Australia	\$4,999,021.84	\$4,999,021.84
Canada	\$1,343,109.10	\$1,343,109.10
France	\$2,490,944.57	\$2,490,944.57
Germany	\$2,775,195.60	\$2,775,195.60
United Kingdom	\$5,057,076.55	\$5,057,076.55
United States	\$9,389,479.79	
Grand Total	\$26,054,827.45	\$16,665,347.67

La siguiente tabla muestra el informe final que podría obtener si usase esta medida en una tabla dinámica:

Non USA Internet Sales	Etiquetas de columna			
Etiquetas de fila	Accessories	Bikes	Clothing	Grand Total
2005		\$1,526,481.95		\$1,526,481.95
2006		\$3,554,744.04		\$3,554,744.04
2007	\$156,480.18	\$5,640,106.05	\$70,142.77	\$5,866,729.00
2008	\$228,159.45	\$5,386,558.19	\$102,675.04	\$5,717,392.68
Total general	\$384,639.63	\$16,107,890.23	\$172,817.81	\$16,665,347.67



# Función RELATEDTABLE

Evalúa una expresión de tabla en un contexto modificado por los filtros dados.

## Sintaxis

RELATEDTABLE (<tableName>)

## Parámetros

Término	Definición
tableName	Nombre de una tabla existente, con la sintaxis estándar de DAX. No puede ser una expresión.

## Valor devuelto

Una tabla de valores.

## Comentarios

Si se han filtrado los datos, la función RELATEDTABLE cambia el contexto en el que se filtran los datos y evalúa la expresión en el nuevo contexto que especifique.

Esta función es un acceso directo para la función CALCULATETABLE sin expresión lógica.

## Ejemplo

El ejemplo siguiente usa la función RELATEDTABLE para crear una columna calculada con Internet Sales en la tabla Product Category.

En la siguiente tabla se muestran los resultados del código mostrado aquí.

Clave de categoría del producto	Valor AlternateKey de categoría del producto	Nombre de categoría del producto	Internet Sales
1	1	Bikes	\$28,318,144.
2	2	Components	
3	3	Clothing	\$339,772.61
4	4	Accessories	\$700,759.96

```
= SUMX( RELATEDTABLE('InternetSales_USD')
, [SalesAmount_USD])
```

# Función VALUES

Devuelve una tabla de una columna que contiene los valores distintos de la tabla o la columna especificada. En otras palabras, los valores duplicados se quitan y solo se devuelven los valores únicos.

Nota
Esta función no se puede usar para devolver valores en una celda o columna de una hoja de cálculo, sino que más bien se usa como una función intermedia, anidada en una fórmula, para obtener una lista de los distintos valores que se pueden contar, o usar para filtrar o sumar otros valores.

## Sintaxis

VALUES (<TableNameOrColumnName>)

## Parámetros

Término	Definición
Tabla o columna	La tabla o la columna cuyos valores únicos se van a devolver.

## Valor devuelto

Columna de valores únicos.

## Comentarios

Al utilizar la función VALUES en un contexto que se ha filtrado, como en una tabla dinámica, los valores únicos devueltos por VALUES se ven afectados por el filtro. Por ejemplo, si se filtra por Region y se devuelve una lista de los valores para City, la lista solo incluirá las ciudades de las regiones permitidas por el filtro. Para devolver todas las ciudades, sin tener en cuenta los filtros existentes, debe utilizar la función ALL para quitar los filtros de la tabla. En el segundo ejemplo se demuestra el uso de ALL con VALUES.

### Funciones relacionadas

En la mayoría de situaciones de ejemplo, los resultados de la función VALUES son idénticos a los de la función DISTINCT. Ambas funciones quitan los duplicados y devuelven una lista de los valores posibles en la columna especificada. Sin embargo, la función VALUES también puede devolver un *Miembro desconocido*. Este valor desconocido es útil cuando se buscan valores distintos de una tabla relacionada, pero un valor usado en la relación no está presente en una tabla. En la terminología de bases de datos, esto se denomina una infracción de la integridad referencial. Tales errores de coincidencia en los datos pueden tener lugar a menudo cuando una tabla se está actualizando y la tabla relacionada no.

La tabla siguiente resume los errores de coincidencia entre datos que pueden producirse en dos tablas relacionadas cuando la integridad referencial no se ha respetado.

Tabla MyOrders	Tabla MySales
June 1	June 1 sales
June 2	June 2 sales
(no se han incluido fechas de pedido)	June 3 sales

Si usó la función DISTINCT para devolver una lista de fechas de la tabla dinámica que contiene estas tablas, solo se devolverán dos fechas. Sin embargo, si usa la función VALUES, la función devuelve las dos fechas más un miembro en

blanco adicional. Asimismo, cualquier fila de la tabla MySales que no tenga una fecha coincidente en la tabla MyOrders se "corresponderá" con este miembro desconocido.

## Ejemplo

La siguiente fórmula cuenta el número de facturas únicas (pedidos de ventas) y genera los siguientes resultados cuando se utiliza en un informe que incluye los nombres de categoría de producto:

Etiquetas de fila	Count Invoices
Accessories	18,208
Bikes	15,205
Clothing	7,461
Total general	27,659

=COUNTROWS (VALUES ( 'InternetSales\_USD' [SalesOrderNumber] ) )