

# Funciones estadísticas

## Función ADDCOLUMNS (DAX)

Agrega columnas calculadas a la tabla o expresión de la tabla especificada.

### Sintaxis

```
ADDCOLUMNS(<table>, <name>, <expression>[, <name>, <expression>]...)
```

### Parámetros

Cualquier expresión de DAX que devuelva una tabla de datos.

Nombre proporcionado a la columna, entre comillas dobles.

Cualquier expresión de DAX que devuelva una expresión escalar, que se evalúa para cada fila de *table*.

### Valor devuelto

Tabla con todas sus columnas originales y las agregadas.

### Ejemplo

En el ejemplo siguiente se devuelve una versión extendida de la tabla Product Category con los valores de ventas totales de los canales distribuidores y de ventas por Internet.

```
ADDCOLUMNS (ProductCategory,
    , "Internet Sales", SUMX (RELATEDTABLE (InternetSales_USD),
    InternetSales_USD [SalesAmount_USD])
    , "Reseller Sales", SUMX (RELATEDTABLE (ResellerSales_USD),
    ResellerSales_USD [SalesAmount_USD]))
```

En la tabla siguiente se muestra una vista previa de los datos que recibiría cualquier función que espere recibir una tabla:

ProductCategory[ProductCategoryName]	ProductCategory[ProductCategoryAlternateKey]	ProductCategory[ProductCategoryKey]	[Internet Sales]	[Reseller Sales]
Bikes	1	1	251077 49.77	630846 75.04
Components	2	2		112058 37.96
Clothing	3	3	306157 .5829	166994 3.267
Accessories	4	4	640920 .1338	534301 .9888

## Función AVERAGE

Devuelve el promedio (la media aritmética) de todos los números de una columna.

## Sintaxis

AVERAGE (<column>)

## Parámetros

Término	Definición
column	Columna que contiene los números para los que desea obtener la media.

## Valor devuelto

Devuelve un número decimal que representa la media aritmética de los números en la columna.

## Comentarios

Esta función toma la columna especificada como un argumento y busca la media de los valores en esa columna. Si desea buscar la media de una expresión que se evalúa como un conjunto de números, use la función AVERAGEX en su lugar.

Los valores no numéricos en la columna se administran como sigue:

- Si la columna contiene texto, no se puede realizar ninguna agregación y las funciones devuelven espacios en blanco.
- Si una columna contiene valores lógicos o celdas vacías, dichos valores se pasan por alto.
- Se incluyen las celdas con valor cero.
- Al calcular el promedio de las celdas, debe tener presente la diferencia entre una celda vacía y una celda que contiene el valor 0 (cero). Cuando una celda contiene 0, se agrega a la suma de números y la fila se cuenta entre el número de filas usado como divisor. Sin embargo, cuando una celda contiene un espacio en blanco, no se cuenta la fila.

Si no hay ninguna fila para agregar, la función devuelve un espacio en blanco. Si hay filas, pero ninguna cumple los criterios especificados, la función devuelve 0. Excel también devuelve cero si no se encuentra ninguna fila que cumpla las condiciones.

## Ejemplo

La fórmula siguiente devuelve la media de los valores en la columna, ExtendedSalesAmount, en la tabla, InternetSales.

=AVERAGE (InternetSales [ExtendedSalesAmount] )

# Función AVERAGEA

Devuelve el promedio (media aritmética) de los valores de una columna. Trata texto y valores no numéricos.

## Sintaxis

AVERAGEA (<column>)

## Parámetros

Término	Definición
column	Columna que contiene los valores para los que desea obtener el promedio.

## Valor devuelto

Un número decimal.

## Comentarios

La función AVERAGEA usa una columna y calcula el promedio de los números que contiene, pero también trata tipos de datos no numéricos según las reglas siguientes:

- Los valores que dan como resultado TRUE cuentan como 1.
- Los valores que dan como resultado FALSE cuentan como 0 (cero).

- Los valores que contienen texto no numérico cuentan como 0 (cero).
- El texto vacío ("") cuenta como 0 (cero)

Si no desea incluir valores lógicos y representaciones de texto de los números en una referencia como parte del cálculo, use la función AVERAGE.

Si no hay ninguna fila para agregar, la función devuelve un espacio en blanco. Si hay filas, pero ninguna cumple los criterios especificados, la función devuelve 0. Microsoft Excel también devuelve cero si no se encuentra ninguna fila que cumpla las condiciones.

## Ejemplo

En el ejemplo siguiente se devuelve el promedio de las celdas que no están en blanco de la columna la que se hace referencia según la tabla siguiente. Si usara la función AVERAGE, el promedio sería 21/2; con la función AVERAGEA, el resultado es 22/5.

Identificador de transacción	Amount	Resultado
0000123	1	Cuenta como 1
0000124	20	Cuenta como 20
0000125	n/d	Cuenta como 0
0000126		Cuenta como 0
0000126	TRUE	Cuenta como 1

=AVERAGEA ([Amount])

# Función AVERAGEX

Calcula el promedio (media aritmética) de un conjunto de expresiones evaluadas sobre una tabla.

## Sintaxis

AVERAGEX(<table>,<expression>)

## Parámetros

Término	Definición
table	Nombre de una tabla o una expresión que especifica la tabla en la que se puede realizar la agregación.
expresión	Expresión con un resultado escalar que se evaluará para cada fila de la tabla en el primer argumento.

## Valor devuelto

Un número decimal.

## Comentarios

La función AVERAGEX permite evaluar las expresiones de cada fila de una tabla y, a continuación, toma el conjunto resultante de valores y calcula su media aritmética. Por consiguiente, la función toma una tabla como su primer argumento y una expresión como segundo argumento.

En el resto de aspectos, AVERAGEX sigue las mismas reglas que AVERAGE. No se pueden incluir las celdas no numéricas o nulas. Se requieren tanto los argumentos de expresión como de tabla.

Si no hay filas para agregar, la función devuelve un espacio en blanco. Si hay filas, pero ninguna cumple los criterios especificados, la función devuelve 0.

## Ejemplo

En el ejemplo siguiente se calculan los impuestos y los cargos medios en cada pedido de la tabla InternetSales sumando en primer lugar Freight y TaxAmt en cada fila y, a continuación, obteniendo el promedio de esas sumas.

```
=AVERAGEX (InternetSales, InternetSales[Freight]+ InternetSales[TaxAmt])
```

Si usa varias operaciones en la expresión usada como segundo argumento, deberá usar paréntesis para controlar el orden de los cálculos. Para obtener más información, vea [Especificación de sintaxis de DAX para PowerPivot](#).

# Función COUNT

La función COUNT cuenta el número de celdas de una columna que contiene números.

## Sintaxis

COUNT (<column>)

## Parámetros

Término	Definición
column	La columna que contiene los números que se van a contar

## Valor devuelto

Un número entero.

## Comentarios

El único argumento permitido para esta función es una columna. Puede usar las columnas que contienen cualquier tipo de datos, pero solo se cuentan los números. La función COUNT cuenta filas que contienen los siguientes tipos de valores:

- Números
- Fechas

Si la fila contiene texto que no se puede traducir en un número, no se cuenta.

Si la función no encuentra ninguna fila para contar, devuelve un espacio en blanco. Si hay filas, pero ninguna cumple los criterios especificados, la función devuelve 0.

## Ejemplo

En el ejemplo siguiente se muestra cómo contar el número de valores numéricos de la columna ShipDate.

```
=COUNT ([ShipDate])
```

Para contar valores lógicos o texto, use las funciones COUNTAX o COUNTA.

# Función COUNTA

La función COUNTA cuenta el número de celdas de una columna que no están vacías. No solo cuenta las filas que contienen valores numéricos, sino también las filas que contienen valores que no estén en blanco, incluido el texto, las fechas y los valores lógicos.

## Sintaxis

COUNTA (<column>)

## Parámetros

Término	Definición
column	La columna que contiene los valores que se van a contar

## Valor devuelto

Un número entero.

## Comentarios

Si no necesita contar celdas que contengan valores lógicos o texto (es decir, si solo desea contar celdas que contienen números), use las funciones COUNTX o COUNT.

Si la función no encuentra ninguna fila para contar, devuelve un espacio en blanco. Si hay filas, pero ninguna cumple los criterios especificados, la función devuelve 0.

## Ejemplo

En el ejemplo siguiente se devuelven todas las filas de la tabla **Reseller** que tienen cualquier tipo de valor en la columna que almacena números de teléfono. Puesto que el nombre de tabla no contiene ningún espacio, las comillas son opcionales.

```
=COUNTA('Reseller'[Phone])
```

# Función COUNTAX

La función COUNTAX cuenta los resultados que no están en blanco al evaluar el resultado de una expresión sobre una tabla. Es decir, funciona igual que la función COUNTA, pero se utiliza para iterar a través de las filas de una tabla y contar las filas donde los resultados de las expresiones especificadas tienen como resultado un resultado que no está en blanco.

## Sintaxis

COUNTAX(<table>,<expression>)

## Parámetros

Término	Definición
table	Tabla que contiene las filas para las que se evaluará la expresión.
expression	Expresión que se va a evaluar por cada fila de la tabla.

## Valor devuelto

Un número entero.

## Comentarios

Como la función COUNTA, la función COUNTAX cuenta las celdas que contienen cualquier tipo de información, incluso otras expresiones.

Por ejemplo, si la columna contiene una expresión que se evalúa en una cadena vacía, la función COUNTAX trata los resultados como resultados que no están en blanco. Normalmente, la función COUNTAX no cuenta las celdas vacías, pero en este caso la celda se cuenta porque contiene una fórmula.

Si no es necesario contar los valores lógicos o de texto, utilice la función COUNTX en su lugar.

Cuando no encuentra ninguna fila para agregar, la función devuelve un valor en blanco. Sin embargo, si hay filas pero ninguna de ellas cumple los criterios especificados, la función devuelve 0. Microsoft Excel también devuelve 0 si no se encuentra ninguna fila que cumpla la condición.

## Ejemplo

El ejemplo siguiente cuenta el número de filas que no están en blanco en la columna Phone mediante la tabla que se obtiene al filtrar la tabla Reseller en [Status] = Active.

```
=COUNTAX(FILTER('Reseller',[Status]="Active"),[Phone])
```

# Función COUNTBLANK

Cuenta el número de celdas en blanco en una columna.

## Sintaxis

COUNTBLANK (<column>)

## Parámetros

Término	Definición
column	La columna que contiene las celdas en blanco que se van a contar.

## Valor devuelto

Un número entero. Si no se encuentra ninguna fila que cumpla la condición, se devuelven espacios en blanco.

## Comentarios

El único argumento permitido para esta función es una columna. Puede usar columnas que contengan cualquier tipo de datos, pero solo se cuentan celdas en blanco. No se cuentan las celdas con el valor cero (0), dado que el cero se considera un valor numérico y no un espacio en blanco.

Si no hay ninguna fila para agregar, la función devuelve un espacio en blanco. Si hay filas, pero ninguna cumple los criterios especificados, la función devuelve 0. Microsoft Excel también devuelve cero si no se encuentra ninguna fila que cumpla las condiciones.

Es decir, si la función COUNTBLANK no encuentra espacios en blanco, el resultado será cero, pero si no hay ninguna fila para comprobar, el resultado estará en blanco.

Esta función de DAX no se puede usar en modo DirectQuery. Para obtener más información acerca de las limitaciones de los modelos DirectQuery, vea <http://go.microsoft.com/fwlink/?LinkId=219172>.

## Ejemplo

En el siguiente ejemplo se muestra cómo contar el número de filas de la tabla Reseller que tienen valores en blanco para BankName.

=COUNTBLANK (Reseller [BankName])

Para contar valores lógicos o texto, use las funciones COUNTAX o COUNTA.

# COUNTROWS (Función) (DAX)

La función COUNTROWS cuenta el número de filas de la tabla especificada o de una tabla definida por una expresión.

## Sintaxis

COUNTROWS (<table>)

## Parámetros

Término	Definición
table	El nombre de la tabla que contiene las filas que se van a contar o una expresión que devuelve una tabla.

## Valor devuelto

Un número entero.

## Comentarios

Esta función se puede utilizar para contar el número de filas de una tabla base, pero se suele usar con más frecuencia para contar el número de filas que es el resultado de filtrar una tabla o de aplicar contexto a una tabla.

Si no hay ninguna fila para agregar, la función devuelve un espacio en blanco. Si hay filas, pero ninguna cumple los criterios especificados, la función devuelve 0. Microsoft Excel también devuelve cero si no se encuentra ninguna fila que cumpla las condiciones.

## Ejemplo

En el ejemplo siguiente se muestra cómo contar el número de filas de la tabla Orders. El resultado esperado es 52761.

```
=COUNTROWS('Orders')
```

En el ejemplo siguiente se demuestra cómo utilizar COUNTROWS con un contexto de fila. En este escenario, hay dos conjuntos de datos que están relacionados por el número de pedido. La tabla Reseller contiene una tabla por cada distribuidor; la tabla ResellerSales contiene varias filas por cada pedido y cada fila contiene un pedido de un distribuidor concreto. Las tablas están conectadas por una relación en la columna ResellerKey.

La fórmula obtiene el valor de ResellerKey y, a continuación, cuenta el número de filas de la tabla relacionada que tienen el mismo identificador de distribuidor. El resultado se envía a la columna CalculatedColumn1.

```
=COUNTROWS(RELATEDTABLE(ResellerSales))
```

En la tabla siguiente se muestra una parte de los resultados esperados:

ResellerKey	CalculatedColumn1
1	73
2	70
3	394

# Función COUNTX

Cuenta el número de filas que contienen un número o una expresión que se evalúa en un número, al evaluar una expresión sobre una tabla.

## Sintaxis

```
COUNTX(<table>,<expression>)
```

## Parámetros

Término	Definición
table	Tabla que contiene las filas que se van a contar.
expression	Una expresión que devuelve el conjunto de valores que contiene los valores que desea contar.

## Valor devuelto

Entero.

## Comentarios

La función COUNTX toma dos argumentos. El primer argumento siempre debe ser una tabla o cualquier expresión que devuelva una tabla. El segundo argumento es la columna o expresión que busca COUNTX. La función COUNTX solo cuenta los valores numéricos o fechas. No se cuentan los argumentos que son valores lógicos ni el texto que no se puede convertir en números. Si la función no encuentra ninguna fila para contar, devuelve un espacio en blanco. Si hay filas, pero ninguna cumple los criterios especificados, la función devuelve 0.

Si desea contar valores lógicos, o texto, use las funciones COUNTAX o COUNTA.

## Ejemplo

La fórmula siguiente devuelve un recuento de todas las filas en la tabla Product que tienen un precio en lista.

```
=COUNTX(Product,[ListPrice])
```

La fórmula siguiente muestra cómo pasar una tabla filtrada a COUNTX para el primer argumento. La fórmula usa una expresión de filtro para obtener solo las filas de la tabla Product que cumplen la condición, ProductSubCategory = "Caps" y, a continuación, cuenta las filas en la tabla resultante que tienen un precio en lista. La expresión FILTER se aplica a la tabla Products, pero usa un valor que se busca en la tabla relacionada, ProductSubCategory.

```
=COUNTX (FILTER (Product, RELATED (ProductSubcategory[EnglishProductSubcategoryName]) = "Caps", Product[ListPrice]))
```

# CROSSJOIN, función (DAX)

Devuelve una tabla que contiene el producto cartesiano de todas las filas de todas las tablas de los argumentos. Las columnas de la nueva tabla son todas las columnas de todas las tablas de argumentos.

## Sintaxis

CROSSJOIN(<table>, <table>[, <table>]...)

## Parámetros

Cualquier expresión de DAX que devuelva una tabla de datos.

## Valor devuelto

Tabla que contiene el producto cartesiano de todas las filas de todas las tablas de los argumentos.

## Comentarios

- Los nombres de columna de los argumentos de *table* deben ser diferentes en todas las tablas o se devuelve un error.
- El número total de filas devueltas por CROSSJOIN() es igual al producto del número de filas de todas las tablas de los argumentos; además, el número total de columnas de la tabla resultante es la suma del número de columnas de todas las tablas. Por ejemplo, si **TableA** tiene **rA** filas y **cA** columnas, **TableB** tiene **rB** filas y **cB** columnas, y **TableC** tiene **rC** filas y **cC** columnas, la tabla resultante tiene **rA × rB × rC** filas y **cA + cB + cC** columnas.

## Ejemplo

En el ejemplo siguiente se muestran los resultados de aplicar CROSSJOIN() a dos tablas: **Colors** y **Stationery**.

La tabla **Colores** contiene colores y patrones:

Color	Patrón
Rojo	Horizontal Stripe
Verde	Vertical Stripe
Azul	Crosshatch

La tabla **Stationery** contiene fuentes y presentación:

Fuente	Presentación
serif	embossed
sans-serif	engraved

La expresión para generar la combinación cruzada se muestra a continuación:

```
CROSSJOIN( Colors, Stationery)
```

Siempre que se use la expresión anterior cuando se espere una expresión de tabla, los resultados de la expresión serán los siguientes:

--	--	--	--



Rojo	Horizontal Stripe	serif	embossed
Verde	Vertical Stripe	serif	embossed
Azul	Crosshatch	serif	embossed
Rojo	Horizontal Stripe	sans-serif	engraved
Verde	Vertical Stripe	sans-serif	engraved
Azul	Crosshatch	sans-serif	engraved

Adiciones de comunidad

# Función DISTINCTCOUNT (DAX)

La función DISTINCTCOUNT cuenta el número de celdas distintas de una columna de números.

## Sintaxis

DISTINCTCOUNT (<column>)

## Parámetros

Columna que contiene los números que se van a contar

## Valor devuelto

Número de valores distintos en *column*.

## Comentarios

El único argumento permitido para esta función es una columna. Puede usar las columnas que contienen cualquier tipo de datos.Cuando la función no encuentra ninguna fila para contar, devuelve BLANK; de lo contrario, devuelve el recuento de valores distintos.

## Ejemplo

En el ejemplo siguiente se muestra cómo contar el número de pedidos distintos en la columna ResellerSales\_USD [SalesOrderNumber].

=DISTINCTCOUNT (ResellerSales\_USD [SalesOrderNumber])

Si se usa la medida anterior en una tabla con el año en el lado y la categoría de producto en la parte superior, se obtienen los siguientes resultados:

Recuento de distintos pedidos a distribuidores	Etiquetas de columna					
Etiquetas de fila	Accessories	Bikes	Clothing	Components		Grand Total
2005	135	345	242	205		366
2006	356	850	644	702		1015
2007	531	1234	963	1138		1521
2008	293	724	561	601		894

					1	1
<b>Grand Total</b>	<b>1315</b>	<b>3153</b>	<b>2410</b>	<b>2646</b>	<b>1</b>	<b>3797</b>

En el ejemplo anterior, el usuario podrá observar que la suma de los números de total general de las filas no es correcta; esto sucede porque en el mismo pedido puede haber elementos de línea, en el mismo orden, de varias categorías de producto.

# GENERATE, función (DAX)

Devuelve una tabla con el producto cartesiano entre cada fila de *table1* y la tabla que resulta de la evaluación de *table2* en el contexto de la fila actual de *table1*.

## Sintaxis

GENERATE(<table1>, <table2>)

## Parámetros

Cualquier expresión de DAX que devuelve una tabla.

Cualquier expresión de DAX que devuelve una tabla.

## Valor devuelto

Una tabla con el producto cartesiano entre cada fila de *table1* y la tabla que resulta de la evaluación de *table2* en el contexto de la fila actual de *table1*.

## Comentarios

- Si la evaluación de *table2* de la fila actual de *table1* devuelve una tabla vacía, la tabla de resultados no contendrá la fila actual de *table1*. Este comportamiento es diferente del de GENERATEALL(), en que la fila actual de *table1* se incluirá en los resultados y las columnas correspondientes a *table2* tendrán valores NULL para esa fila.
- Todos los nombres de columna de *table1* y *table2* deben ser diferentes o se devuelve un error.

## Ejemplo

En el siguiente ejemplo, el usuario desea obtener una tabla de resumen de las ventas por región y categoría de producto del canal de distribuidores, como la tabla siguiente:

SalesTerritory[SalesTerritoryGroup]	ProductCategory[ProductCategoryName]	[Reseller Sales]
Europe	Accessories	\$ 142,227.27
Europe	Bikes	\$ 9,970,200.44
Europe	Clothing	\$ 365,847.63
Europe	Components	\$ 2,214,440.19
North America	Accessories	\$ 379,305.15
North America	Bikes	\$ 52,403,796.85
North America	Clothing	\$ 1,281,193.26
North America	Components	\$ 8,882,848.05
Pacific	Accessories	\$ 12,769.57

Pacific	Bikes	\$ 710,677.75
Pacific	Clothing	\$ 22,902.38
Pacific	Components	\$ 108,549.71

El código siguiente genera la tabla anterior:

```
GENERATE (
SUMMARIZE (SalesTerritory, SalesTerritory[SalesTerritoryGroup])
, SUMMARIZE (ProductCategory
, [ProductCategoryName]
, "Reseller Sales", SUMX (RELATEDTABLE (ResellerSales_USD),
ResellerSales_USD[SalesAmount_USD])
)
)
```

1. La primera instrucción SUMMARIZE, `SUMMARIZE(SalesTerritory, SalesTerritory[SalesTerritoryGroup])`, genera una tabla de grupos de territorios, donde cada fila es un grupo de territorios, como se muestra a continuación:

SalesTerritory[SalesTerritoryGroup]
North America
Europe
Pacific
N/D

2. La segunda instrucción SUMMARIZE, `SUMMARIZE(ProductCategory, [ProductCategoryName], "Reseller Sales", SUMX(RELATEDTABLE(ResellerSales_USD), ResellerSales_USD[SalesAmount_USD]))`, genera una tabla de grupos de categorías de producto con las ventas de los distribuidores de cada grupo, como se muestra a continuación:

ProductCategory[ProductCategoryName]	[Reseller Sales]
Bikes	\$ 63,084,675.04
Components	\$ 11,205,837.96
Clothing	\$ 1,669,943.27
Accessories	\$ 534,301.99

3. Sin embargo, si la tabla anterior se evalúa en el contexto de cada fila de la tabla de grupos de territorios, se obtienen resultados diferentes para cada territorio.

## Función GENERATEALL (DAX)

Devuelve una tabla con el producto cartesiano entre cada fila de *table1* y la tabla que resulta de la evaluación de *table2* en el contexto de la fila actual de *table1*.

# Sintaxis

GENERATEALL(<table1>, <table2>)

## Parámetros

- Cualquier expresión de DAX que devuelve una tabla.
- Cualquier expresión de DAX que devuelve una tabla.

## Valor devuelto

Una tabla con el producto cartesiano entre cada fila de *table1* y la tabla que resulta de la evaluación de *table2* en el contexto de la fila actual de *table1*.

## Comentarios

- Si la evaluación de *table2* de la fila actual de *table1* devuelve una tabla vacía, la fila actual de *table1* se incluirán en los resultados, y las columnas correspondientes a *table2* tendrán valores NULL para esa fila. Este comportamiento es diferente al de GENERATE(), en que la fila actual de *table1***no** se incluirá en los resultados.
- Todos los nombres de columna de *table1* y *table2* deben ser diferentes o se devuelve un error.

## Ejemplo

En el siguiente ejemplo, el usuario desea obtener una tabla de resumen de las ventas por región y categoría de producto del canal de distribuidores, como la tabla siguiente:

SalesTerritory[SalesTerritoryGroup]	ProductCategory[ProductCategoryName]	[Reseller Sales]
Europe	Accessories	\$ 142,227.27
Europe	Bikes	\$ 9,970,200.44
Europe	Clothing	\$ 365,847.63
Europe	Components	\$ 2,214,440.19
N/D	Accessories	
N/D	Bikes	
N/D	Clothing	
N/D	Components	
North America	Accessories	\$ 379,305.15
North America	Bikes	\$ 52,403,796.85
North America	Clothing	\$ 1,281,193.26
North America	Components	\$ 8,882,848.05
Pacific	Accessories	\$ 12,769.57
Pacific	Bikes	\$ 710,677.75
Pacific	Clothing	\$ 22,902.38

Pacific	Components	\$ 108,549.71
---------	------------	---------------

El código siguiente genera la tabla anterior:

```

GENERATEALL (
SUMMARIZE (SalesTerritory, SalesTerritory[SalesTerritoryGroup])
, SUMMARIZE (ProductCategory
, [ProductCategoryName]
, "Reseller Sales", SUMX (RELATEDTABLE (ResellerSales_USD) ,
ResellerSales_USD[SalesAmount_USD])
)
)

```

1. La primera instrucción SUMMARIZE genera una tabla de grupos de territorios, donde cada fila es un grupo de territorios, como los que se muestran a continuación:

SalesTerritory[SalesTerritoryGroup]
North America
Europe
Pacific
N/D

2. La segunda instrucción SUMMARIZE genera una tabla de grupos de categorías de producto con las ventas de los distribuidores de cada grupo, como se muestra a continuación:

ProductCategory[ProductCategoryName]	[Reseller Sales]
Bikes	\$ 63,084,675.04
Components	\$ 11,205,837.96
Clothing	\$ 1,669,943.27
Accessories	\$ 534,301.99

3. Sin embargo, si la tabla anterior se evalúa en el contexto de cada fila de la tabla de grupos de territorios, se obtienen resultados diferentes para cada territorio.

# Función MAX

Devuelve el mayor valor numérico de una columna.

## Sintaxis

MAX (<column>)

## Parámetros

Término	Definición
column	Columna en la que desea buscar el valor numérico más grande.

## Valor de propiedad y valor devuelto

Un número decimal.

### Comentarios

La función MAX toma como argumento una columna que contiene valores numéricos. Si la columna no contiene ningún número, MAX devuelve un espacio en blanco. Si desea evaluar valores que no sean números, utilice la función MAXA.

### Ejemplo

En el ejemplo siguiente se devuelve el valor más grande encontrado en la columna ExtendedAmount de la tabla InternetSales.

```
=MAX (InternetSales [ExtendedAmount])
```

## Función MAXA

Devuelve el valor mayor de una columna. Se cuentan los valores y los blancos.

### Sintaxis

MAXA (<column>)

### Parámetros

Término	Definición
column	Columna en la que desea buscar el valor mayor.

### Valor devuelto

Un número decimal.

### Comentarios

La función MAXA usa una columna como argumento y busca el valor mayor entre los siguientes tipos de valores:

- Números
- Fechas
- Valores lógicos, como TRUE y FALSE. Las filas que se evalúan como TRUE se cuentan como 1 y las filas que se evalúan como FALSE se cuentan como 0 (cero).

Se omiten las celdas vacías. Si la columna no contiene ningún valor que se pueda usar, MAXA devuelve 0 (cero).

Si no desea incluir valores lógicos y espacios en blanco como parte del cálculo, use la función MAX.

### Ejemplo

En el ejemplo siguiente se devuelve el valor mayor de una columna calculada, denominada ResellerMargin, que calcula la diferencia entre el precio de venta y precio del distribuidor.

```
=MAXA ([ResellerMargin])
```

En el ejemplo siguiente se devuelve el valor mayor de una columna que contiene fechas y horas. Por lo tanto, esta fórmula obtiene la fecha de transacción más reciente.

```
=MAXA ([TransactionDate])
```

## función MAXX

Evalúa una expresión por cada fila de una tabla y devuelve el mayor valor numérico.

### Sintaxis

MAXX (<table>, <expression>)

## Parámetros

Término	Definición
table	Tabla que contiene las filas para las que se evaluará la expresión.
expression	Expresión que se va a evaluar por cada fila de la tabla.

## Valor devuelto

Un número decimal.

## Comentarios

El argumento table de la función MAXX puede ser un nombre de tabla o una expresión que se evalúa como una tabla. El segundo argumento indica la expresión que se va a evaluar por cada fila de la tabla.

De los valores que se van a evaluar, solo se cuentan los siguientes:

- Números. Si la expresión no se evalúa como un número, MAXX devuelve 0 (cero).
- Fechas.

Las celdas vacías, los valores lógicos y los valores de texto se omiten. Si desea incluir los valores no numéricos en la fórmula, use la función MAXA.

Si se incluye una celda en blanco en la columna o en la expresión, MAXX devuelve una columna vacía.

## Ejemplo

La siguiente fórmula usa una expresión como el segundo argumento para calcular el importe total de los impuestos y del envío por cada pedido de la tabla InternetSales. El resultado esperado es 375,7184.

```
=MAXX(InternetSales, InternetSales[TaxAmt]+ InternetSales[Freight])
```

La siguiente fórmula primero filtra la tabla InternetSales, mediante una expresión FILTER, para devolver un subconjunto de pedidos de una región de ventas específica, definida como [SalesTerritory] = 5. A continuación, la función MAXX evalúa la expresión usada como el segundo argumento para cada fila de la tabla filtrada y devuelve el importe mayor de impuestos y envíos de esos pedidos únicamente. El resultado esperado es 250,3724.

```
=MAXX(FILTER(InternetSales,[SalesTerritoryCode]="5"), InternetSales[TaxAmt]+  
InternetSales[Freight])
```

# Función MINA

Devuelve el valor menor de una columna, incluidos los valores lógicos y números representados como texto.

## Sintaxis

MINA(<column>)

## Parámetros

Término	Definición
column	La columna para la que desea encontrar el valor mínimo.

## Valor devuelto

Un número decimal.

## Comentarios

La función MINA usa como argumento una columna que contiene números y determina el valor menor de la siguiente forma:

- Si la columna no contiene ningún valor numérico, MINA devuelve 0 (cero).
- Las filas de la columna que se evalúan como valores lógicos, como TRUE y FALSE, se tratan como 1 si es TRUE y 0 (cero) si es FALSE.

- Se omiten las celdas vacías.

Si no desea incluir valores lógicos y texto como parte del cálculo, use en su lugar la función MIN.

## Ejemplo

La expresión siguiente devuelve el cargo de flete mínimo de la tabla InternetSales.

```
=MINA (InternetSales[Freight])
```

La siguiente expresión devuelve el valor mínimo en la columna PostalCode. Como el tipo de datos de la columna es texto, la función no encuentra ningún valor numérico y la fórmula devuelve cero (0).

```
=MINA ([PostalCode])
```

# Función MINX

Devuelve el valor numérico más pequeño que se obtiene al evaluar una expresión para cada fila de una tabla.

## Sintaxis

MINX(<table>, < expression>)

## Parámetros

Término	Definición
table	Tabla que contiene las filas para las que se evaluará la expresión.
expression	Expresión que se va a evaluar por cada fila de la tabla.

## Valor devuelto

Un número decimal.

## Comentarios

La función MINX toma como primer argumento una tabla o una expresión que devuelve una tabla. El segundo argumento contiene la expresión que se evalúa para cada fila de la tabla.

La función MINX evalúa los resultados de la expresión en el segundo argumento según las reglas siguientes:

- Solo se cuentan los números. Si el resultado de la expresión no es un número, MINX devuelve 0 (cero).
- Las celdas vacías, los valores lógicos y los valores de texto se omiten. Los números representados como texto se tratan como texto.

Si desea incluir valores lógicos y representaciones de texto de los números en una referencia como parte del cálculo, use la función MINA.

## Ejemplo

En el ejemplo siguiente solo se filtra la tabla, InternetSales, y solo se devuelven las filas para un zona de ventas determinada. A continuación, la fórmula busca el valor mínimo en la columna Freight.

```
=MINX( FILTER(InternetSales, [SalesTerritoryKey] = 5), [Freight])
```

En el ejemplo siguiente se usa la misma tabla filtrada que en el ejemplo anterior, pero en lugar de buscar meramente los valores en la columna para cada fila de la tabla filtrada, la función calcula la suma de dos columnas, Freight y TaxAmt, y devuelve el valor más pequeño resultante de ese cálculo.

```
=MINX( FILTER(InternetSales, InternetSales[SalesTerritoryKey] = 5),  
InternetSales[Freight] + InternetSales[TaxAmt])
```

## Comentarios

En el primer ejemplo, los nombres de las columnas no son nombres completos. En el segundo ejemplo, los nombres de columna son nombres completos.

# RANK.EQ, función (DAX)



Devuelve la clasificación de un número en una lista de números.

## Sintaxis

`RANK.EQ(<value>, <columnName>[, <order>])`

## Parámetros

Cualquier expresión de DAX que devuelva un único valor escalar cuya clasificación se debe buscar. La expresión se debe evaluar exactamente una vez, antes de que se evalúa la función, y su valor se pasa a la lista de argumentos.

Nombre de una columna existente en la que determinarán las clasificaciones. No puede ser una expresión ni una columna creada mediante estas funciones: `ADDCOLUMNS`, `ROW` or `SUMMARIZE`.

(Opcional) Valor que especifica cómo clasificar *number*, de bajo a alto o de alto a bajo:

value	valor alternativo	Descripción
0 (cero)	FALSE	Clasifica en orden descendente de <i>columnName</i> . Si <i>value</i> es igual al número más alto de <i>columnName</i> , <b>RANK.EQ</b> es 1.
1	TRUE	Clasifica en orden ascendente de <i>columnName</i> . Si <i>value</i> es igual al número más bajo de <i>columnName</i> , <b>RANK.EQ</b> es 1.

## Valor devuelto

Número que indica la clasificación de *value* entre los números de *columnName*.

## Comentarios

- columnName* no puede hacer referencia a ninguna columna creada mediante estas funciones: `ADDCOLUMNS`, `ROW` or `SUMMARIZE.I`
- Si *value* no está en *columnName* o el valor está vacío, *RANK.EQ* devuelve un valor vacío.
- Los valores duplicados de *value* reciben el mismo valor de clasificación. El siguiente valor de clasificación asignado será el valor de posición más el número de valores duplicados. Por ejemplo, si cinco (5) valores tienen la clasificación de 11, el valor siguiente recibirá una clasificación de 16 (11 + 5).

## Ejemplo

En el ejemplo siguiente se crea una columna calculada que clasifica los valores de `SalesAmount_USD`, de la tabla `InternetSales_USD`, en función de todos los números de la misma columna.

```
=RANK.EQ(InternetSales_USD[SalesAmount_USD], InternetSales_USD[SalesAmount_USD])
```

En el siguiente ejemplo se clasifica un subconjunto de valores en función de una muestra dada. Suponga que tiene una tabla de estudiantes locales con el rendimiento en una prueba nacional concreta y, además, tiene el conjunto completo de puntuaciones en dicha prueba nacional. La siguiente columna calculada proporcionará la clasificación nacional de cada uno de los estudiantes locales.

```
=RANK.EQ(Students[Test_Score], NationalScores[Test_Score])
```

# Función ROW (DAX)

Devuelve una tabla con una sola fila que contiene los valores resultantes de las expresiones proporcionadas a cada columna.

## Sintaxis

`ROW(<name>, <expression>[, <name>, <expression>]...)`

## Parámetros

Nombre proporcionado a la columna, entre comillas dobles.

Cualquier expresión de DAX que devuelva un valor escalar único para rellenar. *name*.

## Valor devuelto

Tabla con una sola fila

## Comentarios

Los argumentos siempre deben estar formados por pares de *name* y *expression*.

## Ejemplo

En el ejemplo siguiente se devuelve una tabla de una sola fila con las ventas totales de los canales de Internet y de distribuidores.

```
ROW("Internet Total Sales (USD)", SUM(InternetSales_USD[SalesAmount_USD]),  
    "Resellers Total Sales (USD)", SUM(ResellerSales_USD[SalesAmount_USD]))
```

El código se divide en dos líneas para que sea más fácil leerlo.

# SUMMARIZE (Función) (DAX)

Devuelve una tabla de resumen para los totales solicitados sobre un conjunto de grupos.

## Sintaxis

```
SUMMARIZE(<table>, <groupBy_columnName>[, <groupBy_columnName>]...[, <name>,  
<expression>]...)
```

## Parámetros

Cualquier expresión de DAX que devuelva una tabla de datos.

(Opcional) Nombre completo de una columna existente que se utilizará para crear grupos de resumen en función de los valores situados en ella. Este parámetro no puede ser una expresión.

Nombre dado a una columna de total o de resumen, escrito entre comillas.

Cualquier expresión de DAX que devuelva un único valor escalar, con el que la expresión se va a evaluar varias veces (para cada fila y contexto).

## Valor devuelto

Una tabla con las columnas seleccionadas para los argumentos de *groupBy\_columnName* y las columnas resumidas diseñadas por los argumentos de nombre.

## Comentarios

1. Cada columna en la que se define un nombre debe tener una expresión correspondiente; en caso contrario, se devuelve un error. El primer argumento, nombre, es el nombre de la columna en el resultado. El segundo argumento, expresión, define el cálculo realizado para obtener el valor de cada fila de esa columna.
2. *groupBy\_columnName* debe estar en *table* o en una tabla relacionada con *table*.
3. Cada nombre debe estar entre comillas dobles.
4. La función agrupa un conjunto de filas seleccionado en un conjunto de filas de resumen de acuerdo con los valores de una o más columnas *groupBy\_columnName*. Se devuelve una fila para cada grupo.

## Ejemplo

En el ejemplo siguiente se devuelve un resumen de las ventas de distribuidores, agrupadas en torno al año natural y el nombre de la categoría de producto. Esta tabla de resultados permite realizar análisis sobre las ventas de distribuidor por año y categoría de producto.

```
SUMMARIZE(ResellerSales_USD  
    , DateTime[CalendarYear]  
    , ProductCategory[ProductCategoryName]  
    , "Sales Amount (USD)", SUM(ResellerSales_USD[SalesAmount_USD])  
    , "Discount Amount (USD)", SUM(ResellerSales_USD[DiscountAmount])  
    )
```

En la tabla siguiente se muestra una vista previa de los datos que recibiría cualquier función que espere recibir una tabla:

DateTime[CalendarYear]	ProductCategory[ProductCategoryName]	[Sales Amount (USD)]	[Discount Amount (USD)]
2008	Bikes	12968255.42	36167.6592
2005	Bikes	6958251.043	4231.1621
2006	Bikes	18901351.08	178175.8399
2007	Bikes	24256817.5	276065.992
2008	Componentes	2008052.706	39.9266
2005	Componentes	574256.9865	0
2006	Componentes	3428213.05	948.7674
2007	Componentes	5195315.216	4226.0444
2008	Clothing	366507.844	4151.1235
2005	Clothing	31851.1628	90.9593
2006	Clothing	455730.9729	4233.039
2007	Clothing	815853.2868	12489.3835
2008	Accessories	153299.924	865.5945
2005	Accessories	18594.4782	4.293
2006	Accessories	86612.7463	1061.4872
2007	Accessories	275794.8403	4756.6546

## Opciones avanzadas de SUMMARIZE

### SUMMARIZE con ROLLUP

La adición de la sintaxis ROLLUP() modifica el comportamiento de la función de SUMMARIZE mediante la adición de filas acumuladas al resultado de las columnas de groupBy\_columnName.

#### **Parámetros de ROLLUP**

Nombre completo de una columna existente que se utilizará para crear grupos de resumen en función de los valores situados en ella. Este parámetro no puede ser una expresión.

**Nota:** todos los demás parámetros de SUMMARIZE ya se han explicado y no se repiten aquí por razones de brevedad.

#### **Comentarios**

- No se puede hacer referencia a las columnas mencionadas en la expresión de ROLLUP como parte de una columna de groupBy\_columnName.

## Ejemplo

En el ejemplo siguiente se agregan filas acumuladas a las columnas de Group-By de la llamada a la función SUMMARIZE.

```
SUMMARIZE(ResellerSales_USD
    , ROLLUP( DateTime[CalendarYear], ProductCategory[ProductCategoryName])
    , "Sales Amount (USD)", SUM(ResellerSales_USD[SalesAmount_USD])
    , "Discount Amount (USD)", SUM(ResellerSales_USD[DiscountAmount])
)
```

En la tabla siguiente se muestra una vista previa de los datos que recibiría cualquier función que espere recibir una tabla:

DateTime[CalendarYear]	ProductCategory[ProductCategoryName]	[Sales Amount (USD)]	[Discount Amount (USD)]
2008	Bikes	12968255.42	36167.6592
2005	Bikes	6958251.043	4231.1621
2006	Bikes	18901351.08	178175.8399
2007	Bikes	24256817.5	276065.992
2008	Componentes	2008052.706	39.9266
2005	Componentes	574256.9865	0
2006	Componentes	3428213.05	948.7674
2007	Componentes	5195315.216	4226.0444
2008	Clothing	366507.844	4151.1235
2005	Clothing	31851.1628	90.9593
2006	Clothing	455730.9729	4233.039
2007	Clothing	815853.2868	12489.3835
2008	Accessories	153299.924	865.5945
2005	Accessories	18594.4782	4.293
2006	Accessories	86612.7463	1061.4872
2007	Accessories	275794.8403	4756.6546
2008		15496115.89	41224.3038
2005		7582953.67	4326.4144
2006		22871907.85	184419.1335
2007		30543780.84	297538.0745

		76494758.25	527507.9262
--	--	-------------	-------------

## ROLLUPGROUP

ROLLUPGROUP() se puede usar para calcular grupos de subtotales. Si se usa ROLLUP en contexto, ROLLUPGROUP arrojará el mismo resultado sumando las filas acumuladas al resultado en las columnas groupBy\_columnName. Sin embargo, la adición de ROLLUPGROUP() dentro de una sintaxis de ROLLUP se puede usar para impedir subtotales parciales en las filas acumuladas.

El ejemplo siguiente muestra solo el total de todos los años y categorías sin el subtotal de cada año con todas las categorías:

```
SUMMARIZE(ResellerSales_USD
    , ROLLUP(ROLLUPGROUP( DateTime[CalendarYear],
ProductCategory[ProductCategoryName]))
    , "Sales Amount (USD)", SUM(ResellerSales_USD[SalesAmount_USD])
    , "Discount Amount (USD)", SUM(ResellerSales_USD[DiscountAmount])
)
```

En la tabla siguiente se muestra una vista previa de los datos que recibiría cualquier función que espere recibir una tabla:

<b>DateTime[CalendarYear]</b>	<b>ProductCategory[ProductCategoryName]</b>	<b>[Sales Amount (USD)]</b>	<b>[Discount Amount (USD)]</b>
2008	Bikes	12968255.42	36167.6592
2005	Bikes	6958251.043	4231.1621
2006	Bikes	18901351.08	178175.8399
2007	Bikes	24256817.5	276065.992
2008	Componentes	2008052.706	39.9266
2005	Componentes	574256.9865	0
2006	Componentes	3428213.05	948.7674
2007	Componentes	5195315.216	4226.0444
2008	Clothing	366507.844	4151.1235
2005	Clothing	31851.1628	90.9593
2006	Clothing	455730.9729	4233.039
2007	Clothing	815853.2868	12489.3835
2008	Accessories	153299.924	865.5945
2005	Accessories	18594.4782	4.293
2006	Accessories	86612.7463	1061.4872
2007	Accessories	275794.8403	4756.6546

		76494758.25	527507.9262
--	--	-------------	-------------

## SUMMARIZE con ISSUBTOTAL

Permite al usuario crear otra columna, en la función Summarize, que devuelve TRUE si la fila contiene valores de subtotal para la columna especificada como argumento para ISSUBTOTAL; en caso contrario, devuelve FALSE.

### Parámetros de ISSUBTOTAL

Nombre de cualquier columna de la tabla de la función SUMMARIZE, o de cualquier columna de otra tabla relacionada con ella.

### Valor devuelto

Valor de **True** si la fila contiene un valor de subtotal para la columna proporcionada como argumento; en caso contrario, **False**

### Comentarios

- ISSUBTOTAL solo se puede utilizar en la parte de expresión de una función SUMMARIZE.
- ISSUBTOTAL deben ir precedido de una columna *name* correspondiente.

### Ejemplo

En el ejemplo siguiente se genera una columna de ISSUBTOTAL() para cada una de las columnas de ROLLUP() de la llamada a función SUMMARIZE().

```
SUMMARIZE(ResellerSales_USD
    , ROLLUP( DateTime[CalendarYear], ProductCategory[ProductCategoryName])
    , "Sales Amount (USD)", SUM(ResellerSales_USD[SalesAmount_USD])
    , "Discount Amount (USD)", SUM(ResellerSales_USD[DiscountAmount])
    , "Is Sub Total for DateTimeCalendarYear", ISSUBTOTAL(DateTime[CalendarYear])
    , "Is Sub Total for ProductCategoryName",
ISSUBTOTAL(ProductCategory[ProductCategoryName])
)
```

En la tabla siguiente se muestra una vista previa de los datos que recibiría cualquier función que espere recibir una tabla:

[Is Sub Total for DateTimeCalendarYear]	[Is Sub Total for ProductCategoryName]	DateTime[CalendarYear]	ProductCategory[ProductCategoryName]
FALSE	FALSE		
FALSE	FALSE	2008	Bikes
FALSE	FALSE	2005	Bikes
FALSE	FALSE	2006	Bikes
FALSE	FALSE	2007	Bikes
FALSE	FALSE	2008	Componentes
FALSE	FALSE	2005	Componentes
FALSE	FALSE	2006	Componentes
FALSE	FALSE	2007	Componentes
FALSE	FALSE	2008	Clothing

FALSE	FALSE	2005	Clothing
FALSE	FALSE	2006	Clothing
FALSE	FALSE	2007	Clothing
FALSE	FALSE	2008	Accessories
FALSE	FALSE	2005	Accessories
FALSE	FALSE	2006	Accessories
FALSE	FALSE	2007	Accessories
FALSE	TRUE		
FALSE	TRUE	2008	
FALSE	TRUE	2005	
FALSE	TRUE	2006	
FALSE	TRUE	2007	
TRUE	TRUE		

## Adiciones de comunidad

# Función TOPN (DAX)

Devuelve las N filas superiores de la tabla especificada.

## Sintaxis

TOPN(<n\_value>, <table>, <orderBy\_expression>, [<order>[, <orderBy\_expression>, [<order>]]...])

## Parámetros

Número de filas que se va a devolver. Es cualquier expresión de DAX que devuelva un único valor escalar, con el que la expresión se va a evaluar varias veces (para cada fila y contexto).

Vea la sección de comentarios para averiguar cuándo el número de filas devuelto podría ser mayor que *n\_value*.

Vea la sección notas para saber cuándo se devuelve una tabla vacía.

Cualquier expresión de DAX que devuelve una tabla de datos de donde extraer las “n” filas superiores.

Cualquier expresión de DAX en la que el valor del resultado se utiliza para ordenar la tabla y se evalúa para cada fila de *table*.

(Opcional) Valor que especifica cómo ordenar los valores de *orderBy\_expression*, en orden ascendente o descendente:

value	valor alternativo	Descripción
0 (cero)	FALSE	Ordena en orden descendente de valores de <i>order_by</i> .

		Es el valor predeterminado si se omite el parámetro <i>order</i> .
1	TRUE	Ordena en orden ascendente de valores de <i>order_by</i> .

## Valor devuelto

Una tabla con las N filas superiores de *table* o una tabla vacía si *n\_value* es 0 (cero) o menos. Las filas no se ordenan necesariamente en ningún orden concreto.

## Comentarios

- Si hay valores que coinciden, en los valores de *order\_by* valores, en la fila enésima de la tabla, se devuelven todas las filas con valores coincidentes. Después, si hay valores coincidentes en la enésima fila, la función podría devolver más de n filas.
- Si *n\_value* es 0 (cero) o menos, TOPN devuelve una tabla vacía.
- TOPN no garantiza ningún criterio de ordenación de los resultados.

## Ejemplo

En el ejemplo siguiente se crea una medida con las ventas de los 10 productos más vendidos.

```
=SUMX(TOPN(10, SUMMARIZE(Product, [ProductKey], "TotalSales",
SUMX(RELATED(InternetSales_USD[SalesAmount_USD]), InternetSales_USD[SalesAmount_USD]) +
SUMX(RELATED(ResellerSales_USD[SalesAmount_USD]), ResellerSales_USD[SalesAmount_USD]))
```