

# DAX (I)

## *Introducción al lenguaje DAX v. 2.3*

---

*Jose Ignacio González Gómez*

*Departamento de Economía Contabilidad y Finanzas - Universidad de La Laguna*

[www.jggomez.eu](http://www.jggomez.eu)

### **Introducción**

- Sobre las medidas y el lenguaje DAX
- Entorno de la creación de las medidas.
- Excel y DAX que relación tienen?

### **Bases del lenguaje DAX**

- Tipos de cálculos, columnas y medidas
- Medidas explícitas e implícitas
- Sintaxis

# 1 Introducción

## *Sobre las medidas y el lenguaje DAX*

En nuestros proyectos de BI con Excel (tablas y gráficos dinámicos) o con Power BI necesitamos casi siempre realizar ciertos cálculos para aplicarlos a nuestros informes como es el total de ventas, el coste de las ventas, promedio de facturación por clientes, etc... y a estos cálculos se les conoce como medidas que facilitan la tarea de realizar cálculos en los datos de las tablas que forman parte del modelo mientras se interactúa con las tablas y gráficos dinámicos o elementos visuales y por ello las medidas forman parte del modelo de datos siendo un elemento esencial del mismo.

Es decir, estas acciones o medidas, es el resultado de la aplicación de funciones a los datos del modelo empleando el lenguaje DAX que contiene multitud de funciones similares a las conocidas en Excel ya que DAX se creó a partir del lenguaje Excel y además nos ofrece otras propias como las de Inteligencia de Tiempo

Las funciones DAX están diseñadas para trabajar con datos relacionales y realizar cálculos dinámicos a medida que interactúa con los informes, por tanto, aunque sean similares a las de Excel ***no actúan sobre celdas sino sobre rango de datos condicionados por filtros***

## Entorno para la creacion de medidas en Power Pivot

El proceso general para crear las medidas es a traves de modulo Power Pivot panel cálculos (Medidas – Kpi) opcion nueva medida y acceder ademas al administrador de medidas. Otra opcion de crear las medidas es directamente desde el Power Pivot

Archivos Inicio Insertar Dibujar Disposición de página Fórmulas **Power Pivot**

Administrador Medidas KPI

Agregar a modelo de datos Detectar Configuración

Modelo de datos B4

Nueva medida... Administrar medidas...

Medida

Nombre de la tabla: fDiario

Nombre de la medida: Debe

Descripción: Calcula el total del Debe

Fórmula:  Comprobar fórmula

Asiento	sCPGC	Fecha
1	103	6650000 31/12/20...
2	103	6280000 31/12/20...
4	103	6310000 31/12/20...
5	103	6250000 31/12/20...
6	103	1710000 31/12/20...
7	103	6800000 31/12/20...

Archivos Inicio Diseñar Opciones avanzadas

Pegar Pegar y anexas Pegar y reemplazar Copiar

De base de datos De servicio de datos De otros orígenes Conector existente

Portapapeles Obtener datos externos

[Asiento] fDiario Haber:=Sum(fDiario[THaber])

Asiento	sC...	F...	Documento
1	103	6650000	31/12/2...
2	103	6280000	31/12/2...
3	103	6310000	31/12/2...
4	103	6250000	31/12/2...

Debe: 3346175...

Haber: 3377003...

PGC\_Subcuentas fDiario DPGC\_20071 DEFE dFecha

### Excel y DAX, que relación tienen?

- Mejora de las capacidades de análisis de Excel gracias a la utilización de un nuevo lenguaje de fórmulas llamado DAX especialmente preparado para trabajar con tablas y bases de datos.

# El entorno de trabajo de DAX en Power Pivot

Autoguardado

Archivo Inicio Insertar Disposición Fórmulas

Administrar Medidas KPI Agregar a modelo de datos

Modelo de datos: Nueva medida... Administrar medidas...

B8659

Pegar y anexar Pegar y reemplazar Copiar

De base de datos De servicio de datos De otros orígenes Conexiones existentes

Actualizar Tabla dinámica

Tipo de dato Formato: \$ % For

[Cultivo] fx Agregar columna

Insertar función

Seleccione una categoría:

- Todos
- Fecha y hora
- Matemáticas y trigonometría
- Estadística
- Texto
- Lógico
- Filtro
- Información
- Primario/secundario
- ABS(Número)

Devuelve el valor absoluto de un número.

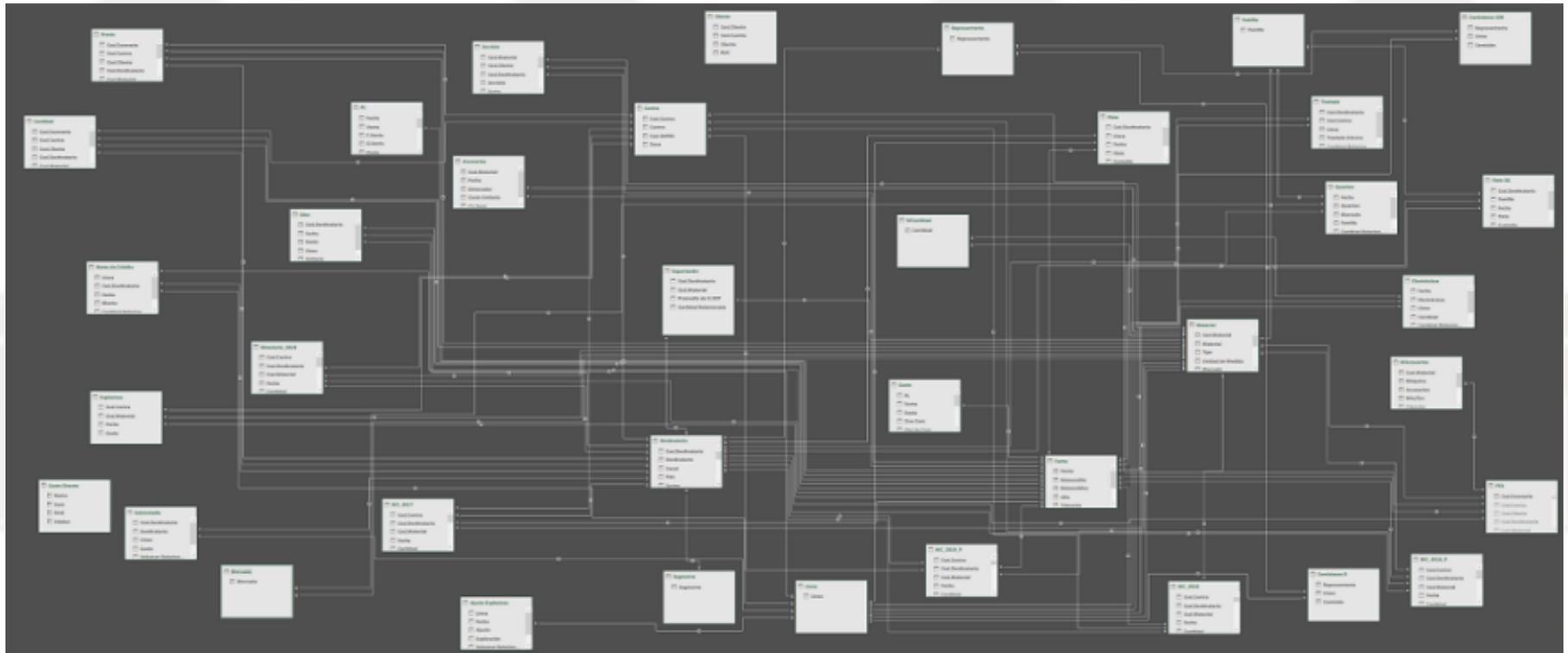
Aceptar Cancelar

	Id	Cultivo	ha	Año	municipio	Tipo Cultivo
1	62	Aceite	0,2	2011	La Matanza	herbaceos
2	90					
3	118					
4	146					
5	174					
6	202					
7	230					
8	258					
9	286					
10	314					
11	342					
12	370					
13	398					
14	426					
15	454					
16	482					
17	510	Aceiga	0,4	2012	La Matanza	herbaceos
18	538	Acelga	1,5	2012	La Orotava	herbaceos

**Podemos crear Medidas o Campos Calculados o Columnas a través de un asistente para generar las funciones**

[Empty cell]

## 2 Bases del lenguaje DAX



Si uno construye un Modelo de Datos adecuado para su modelo de negocio, la cantidad y dificultad de las fórmulas DAX que tendrá que escribir es muy poca. Si por el contrario el Modelo de Datos es malo (léase: poco adecuado para el problema en cuestión), uno tendrá que escribir fórmulas monstruosas y, en consecuencia, ser una especie de «semidiós» del DAX.

A continuación, presentamos una visión general del lenguaje DAX y entre sus características generales podemos destacar las siguientes:

- En DAX no existe el concepto de celda de Excel. DAX siempre hace referencia a tablas y columnas, las funciones esperan tablas o columnas.
- Con DAX podemos crear columnas, campos (medidas) y tablas.
- Una gran parte de las funciones de DAX se denominan igual que las de Excel y tienen el mismo comportamiento, solo cambia la referencia a tablas y columnas.
- DAX cuenta con una tipología de funciones no disponible en Excel fortaleciendo así su potencial como son: Funciones de inteligencia de tiempo, Iteradores., etc.

**Columnas calculadas.** Las características principales de su uso son:

### **Ventajas**

- Pueden ser utilizadas como slicers o segmentadores y en leyendas.
- Se pueden utilizar para definir relaciones entre tablas y agregar a valores.
- Si la tabla no contiene muchos registros y si la fórmula no es compleja son rápidas y fáciles de crear.

### **Inconvenientes**

- Las columnas calculadas se ejecutan durante la actualización de los datos y luego se almacenan en el modelo lo que implica que ocupan espacio en la memoria y por tanto utilizan memoria RAM al estar alojadas en el modelo.
- Si la columna calculada tiene una fórmula compleja, no se recomienda separar el cálculo en columnas intermedias porque cada cálculo intermedio se almacena en la memoria RAM y se desperdicia espacio.

**Las Medidas.** Se recomienda su uso para calcular porcentajes o proporciones, o bien si necesita realizar agregaciones complejas, es decir, cuando se requieren agregar valores calculados con las funciones disponibles en DAX. Presentan las siguientes características:

- *Ventajas*

- ***Utilizadas cuando se necesitan agregar a valores.***
- No tienen peso en el modelo. A diferencia de las columnas las medidas no tienen peso en el modelo, es decir, podemos crear tanta como deseemos y no tendrá impacto en el archivo resultante del análisis

Es decir, esta función o fórmula se empezará a ejecutar y aplicar en el momento en que se añade a la tabla, gráfico o visualización que la contiene. En cambio, la columna calculada no, cuando la tabla se carga al modelo, calcula esa columna se use o no. Cuando el modelo vuelve a actualizarse en una tabla o parte de sus tablas, ellas vuelven a recalcularse o procesarse, mientras que la medida solo actúa solo cuando se aplica o se le llama en una visualización.

- Trabajan en el contexto filtro comúnmente, son calculadas en base a los filtros aplicados en el informe. Así una medida se va adaptando a los selectores y filtros del informe y visualización que le afecte.
- Las medidas también pueden iterar
- Son definidas en una tabla, pero no pertenecen a ella. Es decir, una medida debe ser definida en una tabla pero no pertenece a ella.

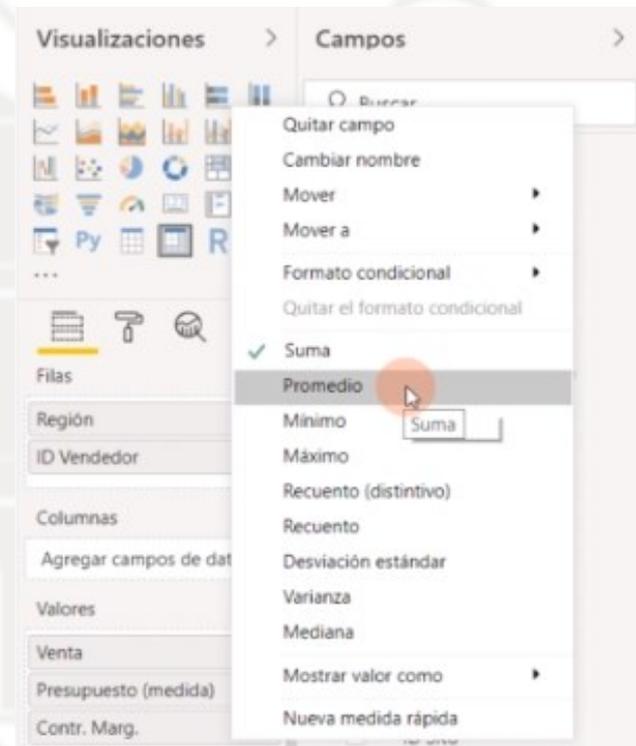
- *Inconvenientes*

- La principal diferencia respecto a las columnas calculadas es el contexto de evaluación.

## Medidas explícitas e implícitas

Las medidas explícitas son las que desarrollamos o creamos directamente con DAX, mientras que las medidas implícitas toman como base una columna simple o calculada de una tabla y se le aplican por defecto la función SUM o bien otras básicas como promedio, máximo, mínimo, Recuento, etc. tal y como se muestra en la ilustración.

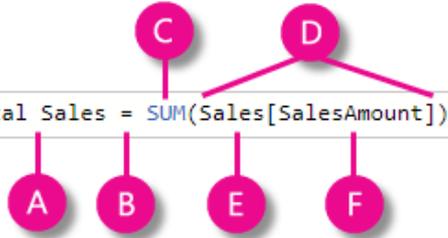
La diferencia entre medidas implícitas y explícitas es que con las primeras no vamos a poder acceder a ellas desde otras partes del modelo, tiene esta importante limitación, sin embargo cuando es una medida explícita si podremos acceder está disponible en nuestro modelo



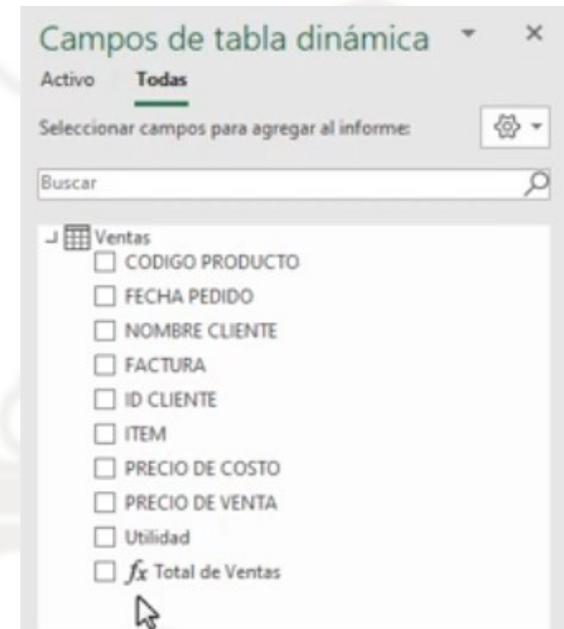
Por tanto, es recomendable siempre utilizar medidas explícitas porque podremos utilizarlas en nuestro modelo y hace que nuestros proyectos sean más robustos.

Señalar que las posibilidades de las medidas implícitas son limitadas tal y como se muestra en la ilustración anterior (sumar, promedio, máximo, mínimo... etc.) mientras con las medidas explícitas contamos con todo un conjunto de funciones que nos van a permitir obtener las mismas que las implícitas más todas las funciones DAX aplicables

## Sintaxis de las fórmulas DAX, elementos de la sintaxis



Nos centraremos en tres conceptos básicos para entender el lenguaje DAX: *sintaxis, funciones y contexto*.



Por ejemplo, puede leer esta fórmula como:

***Para la medida denominada Total Sales, calcular (=) la SUMA de los valores de la columna [SalesAmount] en la tabla Sales.***

Las medidas que vamos creando aparecerán con el índice f(x) en el diseño de nuestras tablas y gráficos.

Esta fórmula incluye los siguientes elementos de sintaxis:

A.El nombre de medida Total Sales.

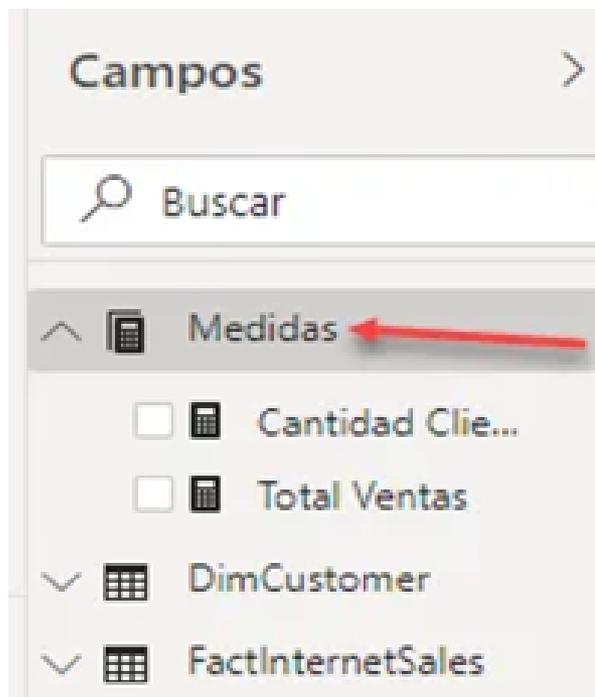
B.El operador signo igual (=) que indica el principio de la fórmula. Al calcular, devolverá un resultado.

C.La función DAX SUM que suma todos los números en la columna Sales[SalesAmount] .

D.Argumentos de la función Los paréntesis () que rodean una expresión que contiene uno o varios argumentos. Todas las funciones requieren al menos un argumento. *Un argumento pasa un valor a una función.*

E.La tabla de referencia Sales.

F.La columna de referencia [SalesAmount] en la tabla Sales. Con este argumento, la función SUM detecta en qué columna se agrega una suma.



Las medidas que vamos creando aparecerán con el indicie  $f(x)$  en el diseño de nuestras tablas, gráficos y visualizaciones o bien con el icono de calculadora en Power BI