Deming
EDITORIAL TECNOLÓGICA

PROGRAMACIÓN DAX





INSTITUTO SUPERIOR TECNOLÓGICO UNIVERSITARIO CORPORATIVO
EDWARDS DEMING®
saber qué hacer y cómo hacerlo

Autor:
JOSÉ LUIS HIDALGO TORRES

José Luis Hidalgo Torres

Programación DAX

Prólogo escrito por el PhD. Carlos Barros

Editorial Deming

Programación DAX
Autor: José Luis Hidalgo Torres

Editorial Deming
Dirección: Calle Edmundo Chiriboga OE-3H y Jorge Páez, Quito, Ecuador

Primera edición – Año 2024

ISBN: 978-9942-692-08-5

Impreso por: GraficArte

Asesoría y colaboración: didáctica editorial

© Este libro es de uso científico, académico e investigativo. Prohibida su comercialización.

Para más información, escribir al correo:
editorial@deming.edu.ec

La programación DAX (Data Analysis Expressions) es esencial para el análisis y la modelización avanzada de datos en entornos como Power BI. Este libro proporciona una guía exhaustiva y técnica sobre los conceptos y aplicaciones de DAX, desde la integración con Power BI hasta la creación y manejo de funciones complejas. En los capítulos iniciales, se cubre la instalación y configuración de Power BI, destacando sus componentes y la preparación del entorno de trabajo. A medida que el lector avanza, se exploran en profundidad las sintaxis de DAX, variables, columnas calculadas y medidas, proporcionando ejemplos prácticos y consejos avanzados para optimizar el uso de DAX. Además, se presentan las diversas funciones de DAX categorizadas por su aplicación, así como las técnicas para crear y administrar relaciones de datos en Power BI. Si pretendes desarrollar competencias avanzadas en el análisis de datos y la programación DAX, este es tu libro.

PhD. Carlos Barros

ÍNDICE DE CONTENIDOS

INTRODUCCIÓN 14
CAPÍTULO UNO POWER BI 17
Introducción 17
¿Qué es Power BI? 17
Componentes del Power BI 18
Instalación de Power BI 18
Paso a paso para descargar Power BI 18
Adaptación de Power BI a su rol 27
El flujo de trabajo en Power BI 28
Bloques de creación de Power BI 29
Visualizaciones 30
Conjuntos de datos 30
Informes 31
Paneles 33
Preparación para obtener los datos introducción a Power Bi Desktop 33
Información general sobre Power BI Desktop 34
Conectarse a datos 34
Limpiar y formatear datos 34
Crear relaciones 35
Limpieza y transformación de datos 38
CAPÍTULO DOS LENGUAJE DAX 42
Introducción 42
¿Qué es DAX? 42
¿Por qué es tan importante DAX? 42
Requisitos previos 43
Contexto 43

Sintaxis de DAX 45
Variables 48
Tablas calculadas 49
Tablas de fechas 49
Análisis de hipótesis 53
Columnas calculadas 55
Creación de columnas calculadas 55
Expresión de la columna calculada 55
Actualización de columnas calculadas 55
Usos de las columnas calculadas 55
Campos calculados o medidas 56
Columnas calculadas vs campos calculados 57
Cálculos 57
Creación de medidas calculadas 59
Medidas rápidas en Power BI 60
Medidas rápidas en Power Pivot 65
Consejos relacionados con la creación de medidas y columnas calculadas 66
Elección entre columnas y medidas calculadas 68
Errores de conversión 70
Errores de operaciones aritméticas 71
Valores vacíos o faltantes 72
Interceptar errores 74
Generando errores 76
Tipos de datos DAX 77
Entero 77
Decimal 77
Divisa 78
Fecha y hora 78

Booleano 78
Cadena 79
Variante 79
Binario 79
Operadores de Dax 79
Aritméticos: 79
Comparación: 80
Lógicos: 81
CAPÍTULO TRES FUNCIONES DAX 82
Introducción 82
Funciones Dax 82
Categoría: Funciones Lógicas 84
Categoría: Funciones Matemáticas 85
Categoría: Funciones Estadísticas 88
Categoría: Funciones de Fecha y Hora 94
Categoría: Funciones de Texto 96
Categoría: Funciones de Filtrado 97
Categoría: Funciones de Inteligencia de Tiempo 99
Categoría: Funciones de Información 101
Categoría: Funciones Primarias y Secundarias 102
Categoría: Otras Funciones 103
CAPÍTULO CUATRO CREAR Y ADMINISTRAR RELACIONES EN POWER BI 104
Introducción 104
Crear una relación de forma manual 105
Editar una relación 107
Propiedades 107
Edición de relaciones mediante diferentes métodos 109
Configuración de más opciones 109

Dirección de filtro cruzado 110
Activar esta relación 111
Descripción de las relaciones 111
Para crear la nueva relación 114
Descripción de las opciones adicionales 115
Actualización de relaciones automáticas 116
Relación activa incorrecta 124
Consulta de todas las relaciones en la vista Relación 127
Búsqueda en el panel Análisis 128
Uso del panel Analytics Con el panel Análisis, puede crear estos tipos de líneas de referencias dinámicas: 129
Trabajo con Vista de datos en Power BI Desktop 133
Ejemplo de aplicación en Power BI 135
GLOSARIO DEL MÓDULO 155
BIBLIOGRAFÍA 156

ÍNDICE DE FIGURAS

Figura 1 Ejemplo de tabla de datos en Excel 31

Figura 2 Ejemplos de informes en Power BI 32

Figura 3 Ventana Obtener datos, de Power BI 37

Figura 4 Vista Informe, en Power BI 38

Figura 5 Transformar datos en Power BI 39

Figura 6 Editor de Power Query 40

Figura 7 Tabla calculada con la función CALENDARAUTO() 50

Figura 8 Relación activa e inactiva entre la tabla Fact Internet Sales y Dim Date. La relación activa relaciona el campo 51

Figura 9 Duplicar tabla Dim Date bajo el nombre Ship Date 52

Figura 10 La tabla Fact Internet Sales está relacionada a través de sus campos OrderDateKey y ShipDateKey con el campo DateKey de la tabla Dim Date y Ship Date respectivamente 52

Figura 11 Menú de creación de un parámetro de hipótesis 54

Figura 12 Tabla calculada a partir de un parámetro de hipótesis 54

Figura 13 Medidas calculadas 59

Figura 14 Sintaxis de medida calculada 60

Figura 15 Operadores de Dax 80

Figura 16 Operadores de comparación 80

Figura 17 Operadores lógicos 81

INTRODUCCIÓN

En este material guía, tiene el propósito de proporcionar los temas de una de las ramas más importantes procesamiento y análisis de datos; se centra el estudio esencialmente en el entorno de Power BI, lenguaje DAX, funciones DAX y la manera de proporcionar los resultados obtenidos en base del procesamiento de los datos.

Capítulo I: El Power BI es una herramienta de análisis empresarial que desempeña un papel crucial en la toma de decisiones y la eficiencia operativa. A continuación, te presento su relevancia:

- Transformación de Datos en Visualizaciones Interactivas:
- Power BI permite convertir grandes volúmenes de datos en informes interactivos que guían la toma de decisiones empresariales.
- Su interfaz intuitiva y amigable facilita la recolección, análisis y visualización de datos provenientes de diversas fuentes, como CRM, Excel y Google Analytics.

Aplicaciones en Diferentes Sectores:

- Profesionales en el sector comercial, desde directores generales hasta expertos en marketing, utilizan los paneles de Power BI.
- En el ámbito de salud, Power BI proporciona datos tanto de los recursos de una clínica como de los pacientes en tiempo real, lo que es esencial para la gestión eficiente y la planificación de necesidades futuras.

Ventajas Clave de Power BI:

- Versatilidad: Se adapta a diversos entornos y necesidades específicas de cualquier empresa.
- Integración: Se conecta fácilmente con sistemas administrativos y otras fuentes de datos.
- Optimización de Decisiones: Ayuda a mejorar informes, detectar nuevas perspectivas de negocio y optimizar recursos.

En resumen, la integración de Power BI en una empresa es vital por la era digital actual. Proporciona la capacidad de consolidar y analizar datos de manera eficiente, facilita la toma de decisiones informadas y fomenta la colaboración entre los miembros del equipo.

Capítulo II: El lenguaje DAX (Data Analysis Expressions) desempeña un papel fundamental en el análisis de datos y la creación de informes interactivos. Su relevancia:

Cálculos Avanzados y Consultas Precisas:

- DAX permite realizar cálculos avanzados en los datos de tablas y columnas relacionadas.
- Facilita la creación de medidas personalizadas que se adaptan al contexto y permiten un análisis más profundo.
- Aplicación en Diferentes Contextos:
- Se utiliza en Analysis Services, Power BI y Power Pivot en Excel.
- Es esencial para crear informes dinámicos y tomar decisiones informadas.

Flexibilidad y Versatilidad:

- DAX abarca funciones, operadores y valores, lo que proporciona flexibilidad en las fórmulas.
- Permite crear nuevos datos a partir de información existente en el modelo.

En resumen, DAX es crucial para el análisis de datos y la generación de insights significativos. Explorar su potencial puede impulsar la eficiencia y la toma de decisiones en cualquier organización.

Capítulo III: Las funciones DAX (Data Analysis Expressions) son esenciales para el análisis de datos y la creación de informes interactivos.

DAX es un lenguaje de expresiones de fórmulas utilizado en Analysis Services, Power BI y Power Pivot en Excel.

Estas funciones abarcan funciones, operadores y valores para realizar cálculos avanzados y consultas en los datos de las tablas y columnas relacionadas de los modelos de datos tabulares.

Importancia de las Funciones DAX:

- Cálculos Avanzados: DAX permite realizar cálculos complejos en los datos, como sumas, promedios, conteos, etc.
- Medidas Personalizadas: Con DAX, puedes crear medidas personalizadas que se adaptan al contexto y permiten un análisis más profundo en informes y dashboards.
- Optimización de Consultas: Las funciones DAX son fundamentales para optimizar el rendimiento de los informes y garantizar consultas precisas.

Aplicación en Diferentes Contextos:

- Desde informes empresariales hasta análisis de salud, DAX se utiliza en diversos sectores para tomar decisiones informadas.
- Su versatilidad y flexibilidad lo convierten en una herramienta poderosa para cualquier profesional que trabaje con datos.

En resumen, las funciones DAX son un componente clave para desbloquear el potencial de tus datos y generar insights significativos. Explorar y dominar DAX puede marcar la diferencia en tu trabajo analítico.

CAPÍTULO UNO

POWER BI

Introducción

El tratamiento y visualización de datos es una necesidad para todo tipo de empresas. Power BI ofrece una suite de herramientas que facilitan la creación de consultas, informes y cuadros de mando con alto nivel de interactividad y dinamismo.

Según (Quantec, s.f.):

¿Qué es Power BI?

Power BI es una plataforma de inteligencia empresarial (BI) unificada y escalable que permite la visualización y conexión de datos de diversas fuentes, transformándolos en información coherente, interactiva y atractiva visualmente. Power BI incluye una colección de servicios de software, aplicaciones y conectores que trabajan juntos para lograr este objetivo. La plataforma es capaz de conectarse a cualquier fuente de datos, ya sea una simple hoja de cálculo de Excel o una colección de almacenes de datos híbridos locales o basados en la nube. Power BI también permite la fácil visualización y descubrimiento de información importante, así como su compartición con cualquier persona.

Power BI es sencillo y rápido; capaz de crear información a partir de una hoja de cálculo de Excel o una base de datos local. También es estable y tiene una funcionalidad apta para empresas, listo para un modelado exhaustivo y un análisis en tiempo real, así como para un desarrollo personalizado. Puede transformarse en su herramienta personal para la creación de informes y visualización, interactúa con el motor de análisis y de decisión para impulsar proyectos en grupo, divisiones o empresas enteras.

Componentes del Power BI

Power BI consta de tres elementos principales:

- **Power BI Desktop**: una aplicación de escritorio para Windows que permite a los usuarios crear informes personalizados y paneles interactivos.
- **Power BI Service**: un servicio SaaS (software como servicio) en línea que permite a los usuarios compartir informes y paneles con otros usuarios.
- **Power BI Mobile**: aplicaciones móviles disponibles para teléfonos y tabletas Windows, así como para dispositivos iOS y Android, que permiten a los usuarios acceder a informes y paneles desde cualquier lugar.

Estos tres elementos están diseñados para permitir a los usuarios crear, compartir y utilizar información empresarial de la forma que les resulte más eficaz para su rol.

Instalación de Power BI

Según (G-Talent.net, 2023):

> Power BI es un servicio de análisis empresarial que ofrece Microsoft, que da la oportunidad de conectar los datos y visualizarlos a través de la plataforma unificada de manera interactiva, ayuda a elaborar modelos de datos eficientes e informes asombrosos para mejorar las capacidades de gestión empresarial.

Al crear un archivo de Power BI Desktop propio para publicarlo en el servicio Power BI y compartirlo con otros usuarios. En este artículo te indicaremos el paso a paso para descargar e instalar Power BI en tu ordenador de forma sencilla.

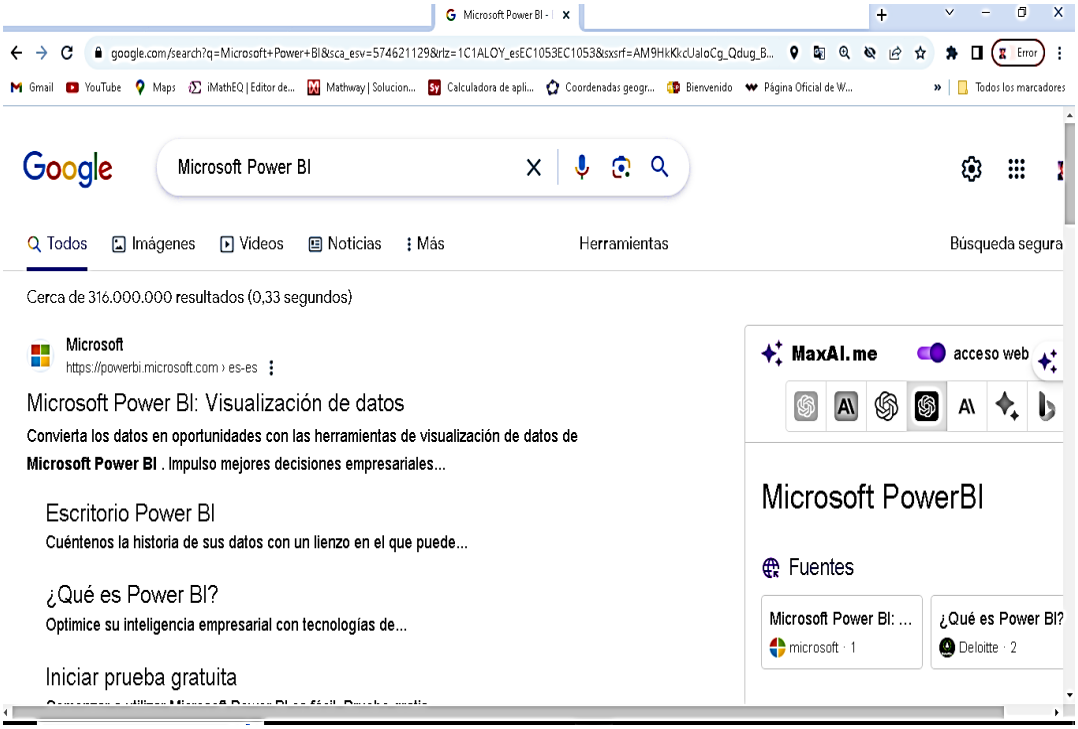
Paso a paso para descargar Power BI

El primer paso, es ingresar al sitio web de Power BI:

Escriba en Google: Microsoft Power BI (Figura 1)

Figura **1**

Página web de Google

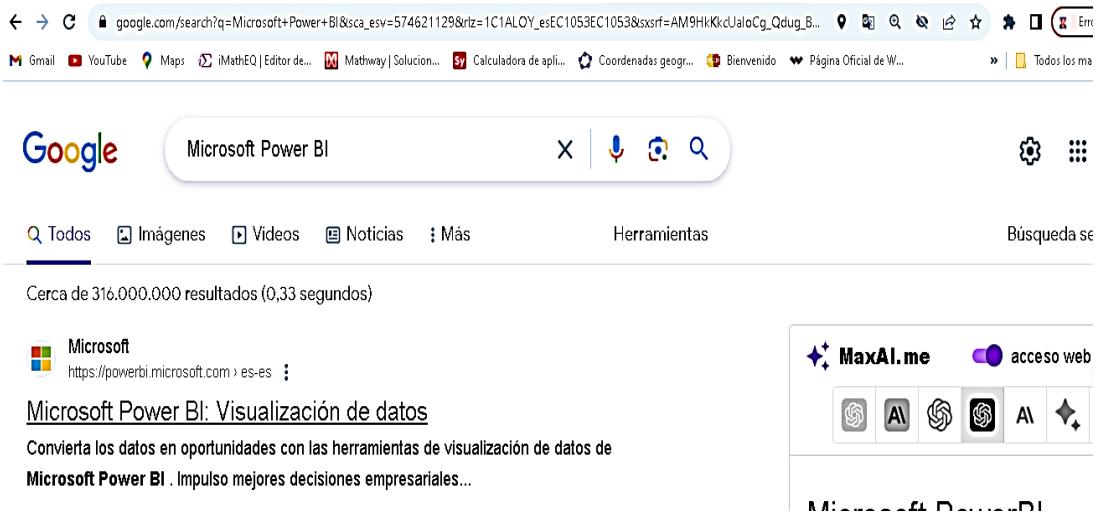


Fuente: Internet

Se da un clic en: Microsoft Power BI: Visualización de datos (Figura 2)

Figura 2

Página web de Google: Microsoft Power BI: Visualización de datos



Fuente: Internet

Después de dar clic en Microsoft Power BI: Visualización de datos (Figura 3), se tiene:

Figura 3

Google: Visualización de datos

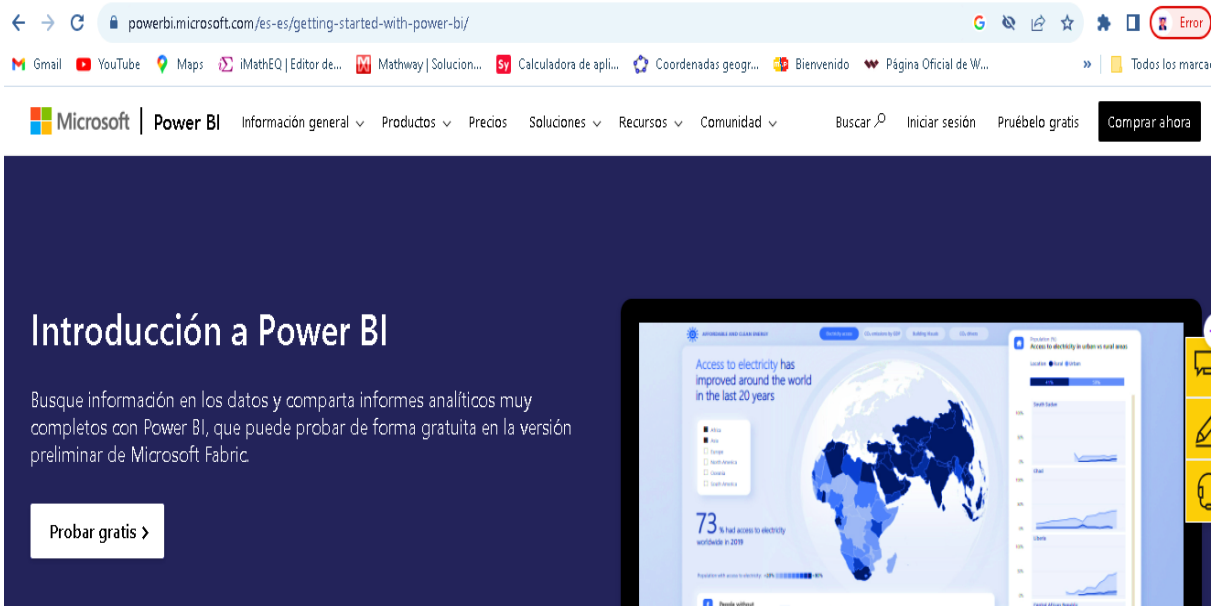


Fuente: Internet

En Iniciar prueba gratuita, se da clic (Figura 4):

Figura 4

Introducción a Power BI

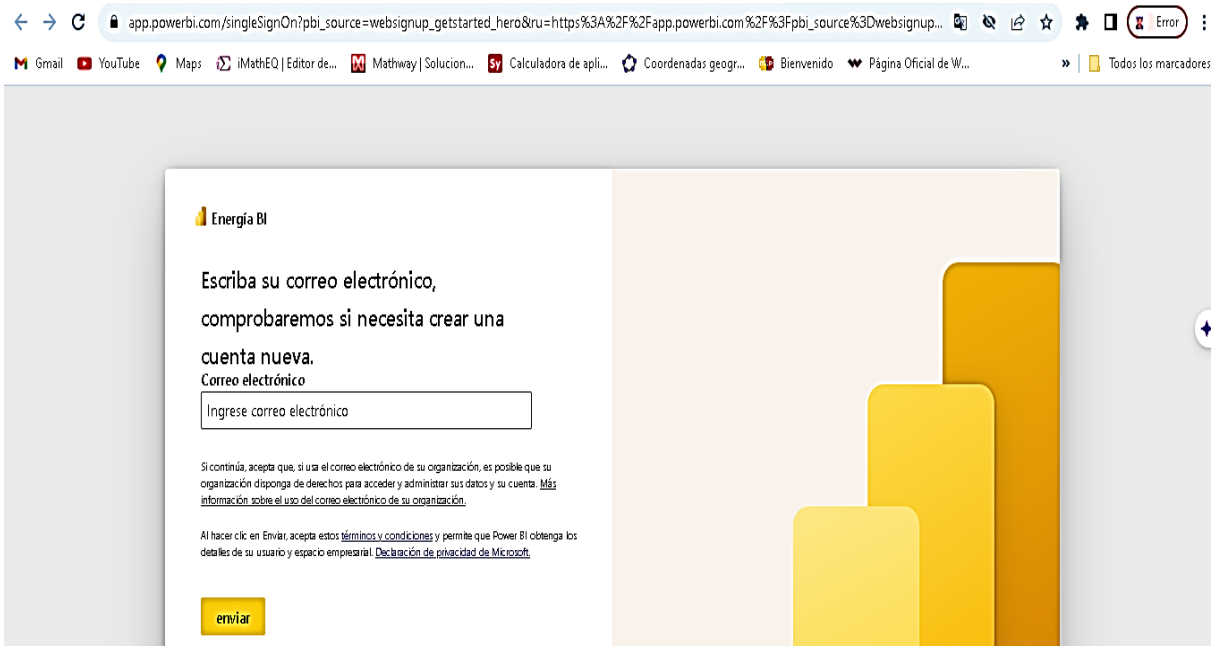


Fuente: Internet

Se da clic en Probar gratis (Figura 5):

Figura 5

Iniciar sesión de Power BI



Fuente: Google

Te pide un correo, que podría ser el personal institucional; colocas el correo y se da clic en enviar.

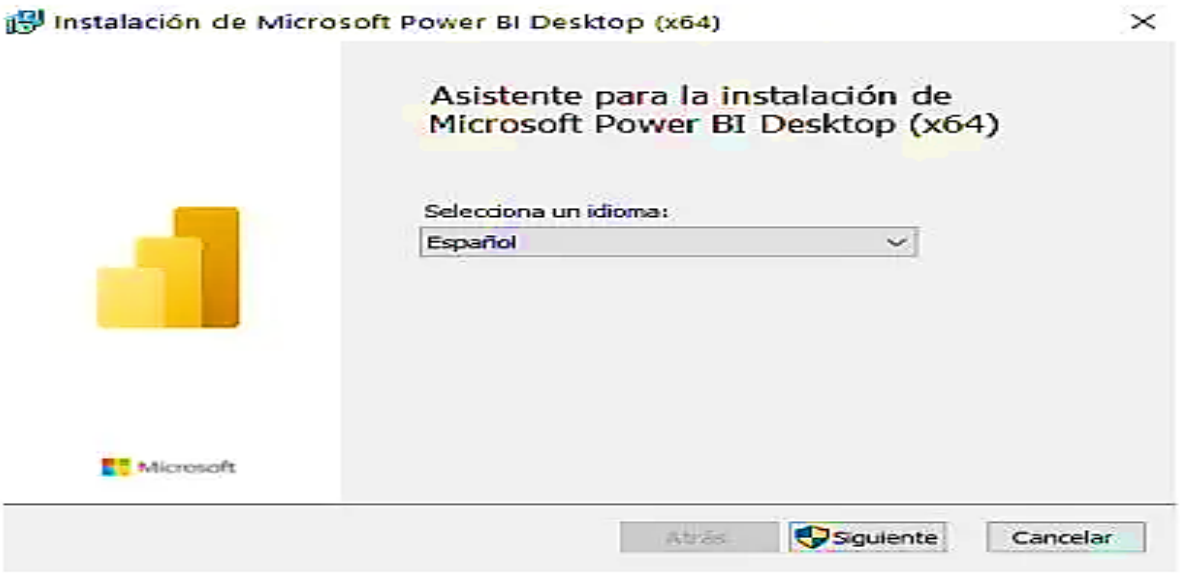
A tu correo te envían un código, que deberá ser colocado en la próxima ventana y continuar con las instrucciones que te va dando el paquete de instalación.

A continuación, ya bajado el software de instalación, se aplican los siguientes pasos: Al estar descargado el archivo en el ordenador, se debe hacer doble clic en el archivo para iniciar la instalación de Power BI.

Se abrirá la siguiente ventana.

Figura 6

Instalación de Power BI



Fuente: Google

Escoja el idioma deseado y presione en el botón “Siguiente”.

Se abrirá la siguiente ventana (Figura 7).

Figura 7

Ventana de Asistente de instalación de Power BI



Fuente: Google

Haga clic en el botón siguiente.

Posteriormente, se abrirá la ventana de los términos de licencia del Software. Selecciones “Acepto los términos del contrato de licencia” (Figura 8).

Figura 8

Términos de la licencia de Power BI



Fuente: Google

Haga clic en siguiente.

Al continuar deberá seleccionar la carpeta destino (Figura 9) para la instalación.

Figura 9

Carpeta destino del software de Power BI



Fuente: Google

Haga clic en siguiente.

Casi termina la instalación (Figura 10), solo debes hacer clic en el botón Instalar.

Figura 10

Paso final para la instalación de Power BI



Fuente: Google

Al hacerlo, iniciará el proceso de instalación del software de Power BI (Figura 11).

Figura 11

Proceso de instalación



Fuente: Google

Una vez completada la instalación (Figura 12), se mostrará la siguiente pestaña.

Figura 12

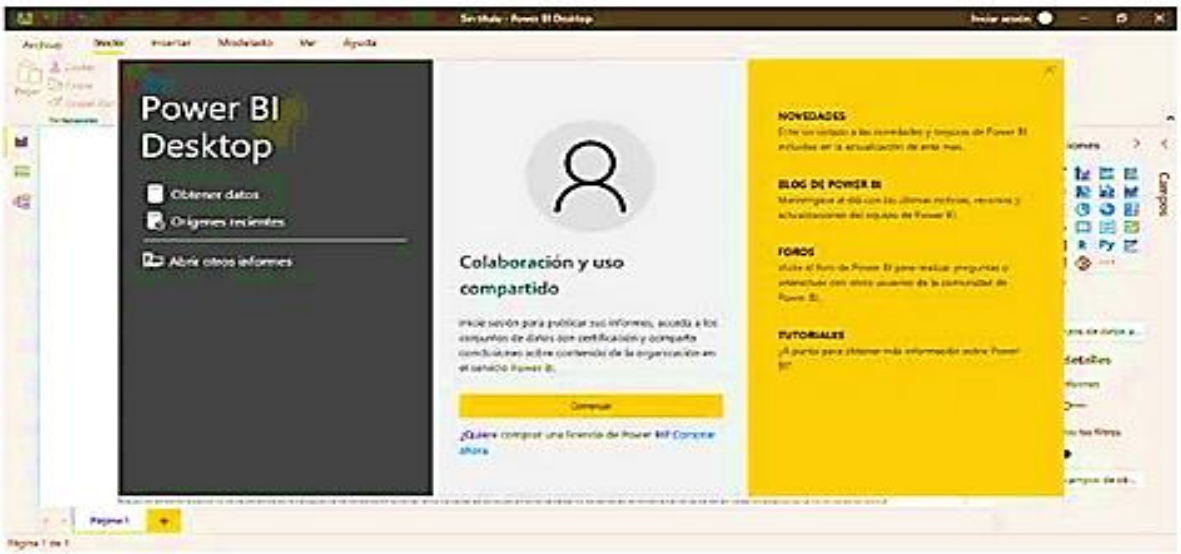
Completación del proceso de instalación del Power BI



Fuente: Google

Haga clic en el botón finalizar para abrir el software Power BI para Desktop. Ya está instalado en tu ordenador el Software.

Cuando abra Power BI, podrá observar la siguiente información en la ventana.



Al iniciar en Power BI Desktop por primera vez, es decir, el proceso de instalación no es una actualización, será necesario completar un formulario con su información personal o que inicie sesión en el servicio para poder seguir. Al iniciar sesión podrá publicar sus informes y acceder a todos los beneficios que ofrece Power BI.

Si ya posee una cuenta en Power BI haga clic en el botón Comenzar, posteriormente ingrese su correo electrónico para iniciar sesión y haga clic en continuar. Si aún no ha adquirido una licencia, haga clic en Comprar ahora para obtener una licencia de Power BI.



Nota: Para la instalación de Power BI, podrían visualizarlo también en la dirección de YouTube: https://www.youtube.com/watch?v=yCFNSg4EWpc

Adaptación de Power BI a su rol

La forma en que se utilice Power BI puede variar en función de su rol en un proyecto o equipo.

Power BI ofrece tres elementos principales: Power BI Desktop, el servicio Power BI y las aplicaciones para dispositivos móviles. Cada uno de estos elementos tiene sus propias ventajas y se adapta a diferentes necesidades.

Por ejemplo, Power BI Desktop es la herramienta más potente para crear informes y paneles personalizados. Se utiliza a menudo por analistas de datos y profesionales de inteligencia empresarial. El servicio Power BI es una plataforma en línea que permite compartir informes y paneles con otros usuarios. Es una buena opción para equipos de trabajo que necesitan colaborar en el análisis de datos. Las aplicaciones para dispositivos móviles permiten a los usuarios acceder a sus informes y paneles desde cualquier lugar. Son una buena opción para los usuarios que necesitan realizar análisis de datos de forma rápida y sencilla.

Un analista de datos podría utilizar Power BI Desktop para crear un informe que compare los resultados de ventas de diferentes productos. Un director de ventas podría

utilizar el servicio Power BI para crear un panel que muestre el progreso de las cuotas de ventas. Un gerente de producción podría utilizar las aplicaciones para dispositivos móviles de Power BI para supervisar el inventario en tiempo real.

Power BI ofrece una serie de beneficios que lo convierten en una herramienta valiosa para el análisis de datos. Estos beneficios incluyen:

- **Facilidad de uso**: Power BI es una herramienta intuitiva que es fácil de aprender y usar.
- **Flexibilidad**: Power BI ofrece una amplia gama de funciones y capacidades que se adaptan a diferentes necesidades.
- **Integridad**: Power BI se basa en la plataforma Microsoft Azure, que ofrece una gran seguridad y confiabilidad.

El flujo de trabajo en Power BI

Un flujo usual de trabajo en Power BI, consta de los siguientes pasos:

- **Creación de un informe en Power BI Desktop**: En este paso, el usuario utiliza Power BI Desktop para conectarse a los datos, limpiarlos y transformarlos, y crear el informe.
- **Conexión a los datos**: El usuario puede conectarse a una amplia gama de orígenes de datos, como bases de datos, archivos y aplicaciones web.
- **Limpieza y transformación de los datos**: El usuario puede limpiar y transformar los datos para que sean más fáciles de analizar.
- **Creación del informe**: El usuario puede utilizar una amplia gama de objetos visuales para crear informes personalizados.
- **Publicación del informe en el servicio Power BI**: En este paso, el usuario publica el informe en el servicio Power BI para que otros usuarios puedan acceder a él.
- **Publicación**: El usuario puede publicar el informe en el servicio Power BI de forma privada o pública.

- **Acceso al informe**: Los usuarios pueden acceder al informe desde cualquier dispositivo con conexión a Internet.
- **Compartición del informe**: En este paso, el usuario comparte el informe con otros usuarios.
- **Compartición**: El usuario puede compartir el informe con otros usuarios a través de correo electrónico, enlaces o aplicaciones móviles.
- **Acceso al informe**: Los usuarios pueden acceder al informe desde cualquier dispositivo con conexión a Internet.

No siempre tendrá que realizarse de esa forma, pero es un método fácil de seguir. Para comenzar, lo primero que vamos a explicar son los bloques de creación básicos de Power BI; para obtener una base sólida que convierta a los datos en informes y objetos visuales atractivos.

Bloques de creación de Power BI

Todo lo que se haga en Power BI, se puede dividir en unos pocos bloques de creación básicos. Una vez comprendido, podrá partir de cada uno de ellos para crear informes elaborados y complejos.

Los bloques de creación básicos de Power BI, son:

- **Visualizaciones**: Representaciones visuales de los datos, como gráficos, tablas, mapas y diagramas.
- **Conjuntos de datos**: Conjuntos de datos estructurados que se utilizan para crear informes y paneles.
- **Informes**: Páginas que contienen visualizaciones que se utilizan para presentar datos.
- **Paneles**: Conjuntos de visualizaciones que se utilizan para proporcionar una vista general de los datos.

De manera similar como los objetos complejos se construyen a partir de bloques de creación básicos; los informes y paneles de Power BI se construyen a partir de estos cuatro bloques de creación básicos.

Visualizaciones

Las visualizaciones son la forma en que se presentan los datos en Power BI. Pueden utilizarse para mostrar una amplia gama de información, desde tendencias históricas hasta relaciones entre diferentes datos.

Las visualizaciones son una forma de representar los datos de forma visual. Pueden utilizarse para mostrar una amplia gama de información, desde tendencias históricas hasta relaciones entre diferentes datos. Power BI ofrece una amplia gama de tipos de visualizaciones, desde gráficos básicos hasta visualizaciones más complejas.

Las visualizaciones pueden ser sencillas, como un único número que representa un aspecto significativo, o visualmente complejas, como un mapa de colores degradados que muestra la opinión del votante con respecto a un determinado problema o preocupación social. La finalidad de una visualización es presentar los datos de una manera que ofrezca contexto e información detallada, lo que probablemente resultaría difícil de discernir en una tabla sin formato de números o texto.

Los beneficios de utilizar visualizaciones en Power BI incluyen una mejor comprensión de los datos, una mejor comunicación de los datos y un mayor impacto en los usuarios.

Conjuntos de datos

Los conjuntos de datos son la fuente de datos que se utilizan para crear visualizaciones. Pueden conectarse a una amplia gama de orígenes de datos, como bases de datos, archivos y aplicaciones web.

Los conjuntos de datos son la fuente de datos que se utilizan para crear visualizaciones en Power BI. Pueden conectarse a una amplia gama de orígenes de datos, como bases de datos, archivos y aplicaciones web.

Los conjuntos de datos pueden ser sencillos, como una tabla de Excel Figura 13, o complejos, como una combinación de datos de diferentes fuentes. El propósito de un conjunto de datos es proporcionar una colección de datos que pueda ser utilizada para crear visualizaciones.

Figura 13

Ejemplo de tabla de datos en Excel



	A	B	C	D	E	F	G
1	Matrícula	Nombre	Apellidos	Puesto	Estado	Antigüedad	Sueldo Quincenal
2	248	Margarita	Duque Juarez	Administrador	D.F.	7	$ 12,000.00
3	85	Nelly	Zambada	Director	San Luis Potosi	28	$ 10,000.00
4	105	Francisco	Nájera Tápia	Medico Familiar	Puebla	8	$ 10,000.00
5	115	Hugo	Loredo Soto	Medico Familiar	Oaxaca	19	$ 10,000.00
6	131	Rosa	Mendez Ortega	Medico Familiar	Oaxaca	1	$ 10,000.00
7	132	Victor	Mosso Guzman	Medico Familiar	Puebla	1	$ 10,000.00
8	136	Rosa	Ruíz Dávila	Director	D.F.	28	$ 10,000.00

Fuente: (Quantec, s.f.)

Los conjuntos de datos se pueden crear de varias maneras. Una forma es conectar Power BI a un origen de datos existente, como una base de datos o una tabla de Excel. Otra forma es crear un conjunto de datos desde cero, utilizando la función "Crear conjunto de datos".

Los conjuntos de datos ofrecen una serie de beneficios, entre los que se incluyen la flexibilidad, la eficiencia y la accesibilidad.

Informes

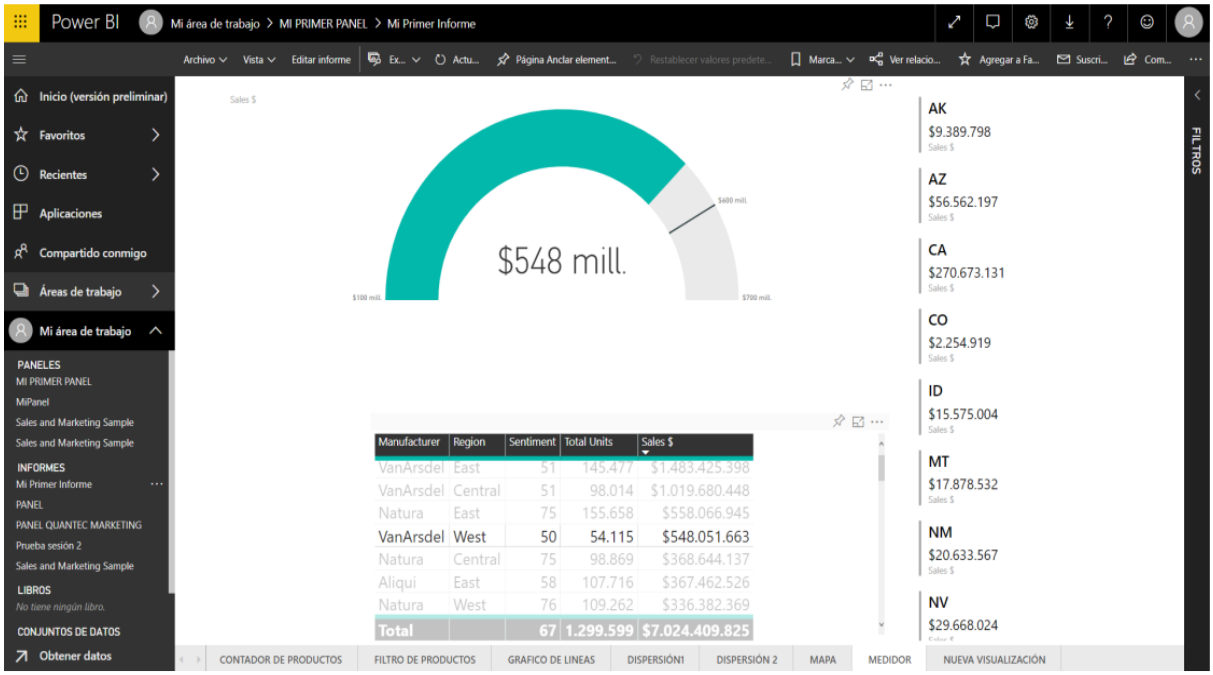
Los informes son páginas que contienen visualizaciones. Se pueden utilizar para presentar datos de forma organizada y atractiva.

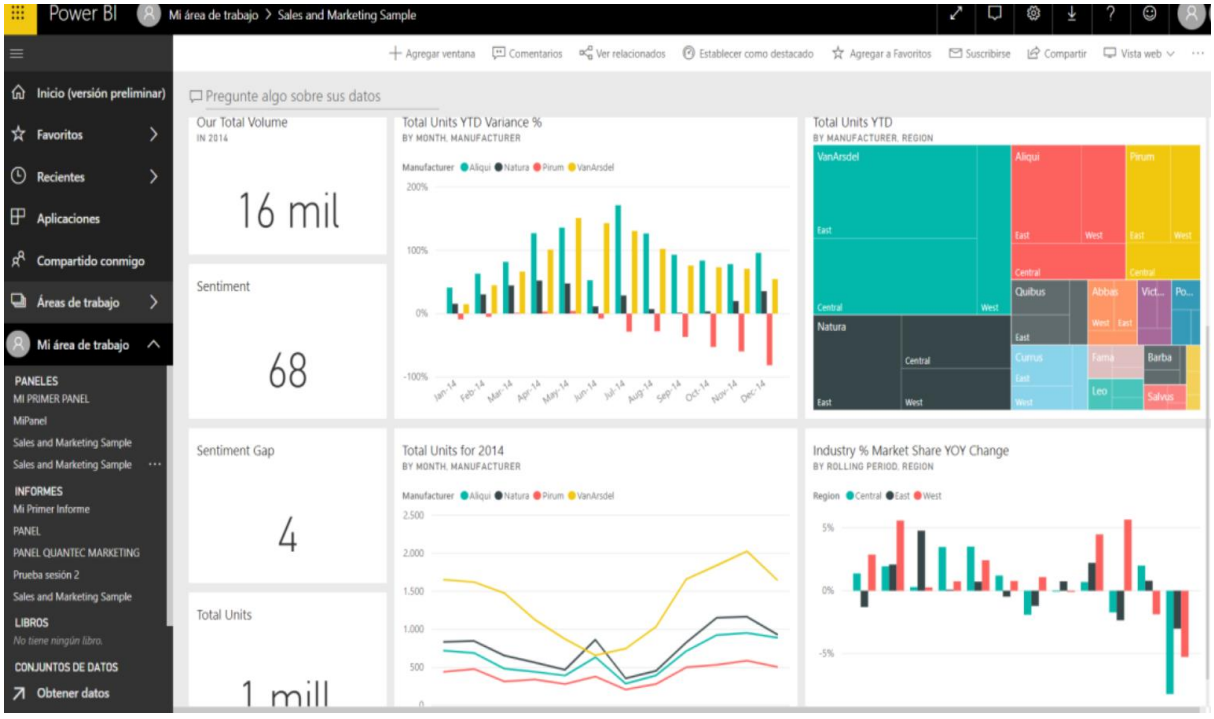
En Power BI, un informe es una colección de visualizaciones que aparecen juntas en una o varias páginas. Los informes son una forma de presentar los datos de forma organizada y atractiva.

Los informes pueden ser sencillos, como un informe (Figura 14) que muestra las ventas de un producto en una sola página, o complejos, como un informe que muestra datos de diferentes fuentes en varias páginas.

Figura 14

Ejemplos de informes en Power BI





Fuente: (Quantec, s.f.)

Los informes se pueden crear de varias maneras. Una forma es crear un informe en blanco y añadir visualizaciones. Otra forma es utilizar un modelo de informe prediseñado. Los informes ofrecen una serie de beneficios, entre los que se incluyen la mejor comprensión de los datos, la mejor comunicación de los datos y el mayor impacto en los usuarios.

Paneles

Los paneles son conjuntos de visualizaciones que se utilizan para proporcionar una vista general de los datos.

Los paneles de Power BI son colecciones de visualizaciones de una sola página que se pueden compartir con otros usuarios. Son una forma eficaz de compartir información importante con un público amplio.

Los paneles se pueden utilizar para una variedad de propósitos, como proporcionar una visión general de los datos, mostrar tendencias o destacar indicadores clave de rendimiento (KPI).

Los paneles se pueden crear de varias maneras. Una forma es crear un panel en blanco y añadir visualizaciones. Otra forma es utilizar un modelo de panel prediseñado. Los paneles ofrecen una serie de beneficios, entre los que se incluyen la compartición, la personalización y la visibilidad.

Los bloques de creación básicos de Power BI, son una herramienta muy útil, que puede utilizarse para crear informes y paneles personalizados. Al comprender cómo funcionan estos bloques de creación básicos, podrá crear informes y paneles que sean informativos y atractivos.

Preparación para obtener los datos introducción a Power Bi Desktop

En esta sección se examinarán las herramientas y características basadas en los datos de Power BI, con especial hincapié en Power BI Desktop. Muchas de estas herramientas también se aplican al servicio Power BI, por lo que esta sección tiene una doble función.

La sección se centra en las siguientes herramientas y características de Power BI:

- **Conexión a datos**: Power BI Desktop ofrece una amplia gama de conectores de datos que permiten conectarse a una amplia gama de orígenes de datos, como bases de datos, archivos y aplicaciones web.
- **Limpieza de datos**: Power BI Desktop ofrece una serie de herramientas y características que permiten limpiar datos, como eliminar valores duplicados, corregir errores y transformar datos.
- **Visualización de datos**: Power BI Desktop ofrece una amplia gama de visualizaciones que permiten representar datos de forma visual.

Power BI Desktop es una herramienta potente y versátil que permite obtener, limpiar y visualizar datos de forma rápida y sencilla. Es una herramienta esencial para cualquier usuario que trabaje en proyectos de inteligencia empresarial.

Información general sobre Power BI Desktop

Power BI Desktop es una herramienta de análisis de datos, que permite a los usuarios conectarse a una amplia variedad de orígenes de datos, limpiarlos y aplicarles formato, y crear relaciones entre ellos. Esta herramienta es ideal para usuarios de todos los niveles de experiencia, ya que ofrece una interfaz intuitiva y un conjunto de funciones completo.

Conectarse a datos

Power BI Desktop admite una amplia variedad de orígenes de datos, incluidos archivos locales, bases de datos, servicios en la nube y aplicaciones SaaS. Para conectarse a un origen de datos, simplemente seleccione el tipo de origen de datos y siga las instrucciones.

Limpiar y formatear datos

Una vez que haya conectado sus datos, Power BI Desktop le permite limpiarlos y aplicarles formato para que resulten más útiles. Puede dividir columnas, cambiarles el nombre, cambiar los tipos de datos y trabajar con fechas.

Crear relaciones

Las relaciones entre columnas permiten a Power BI Desktop unir datos de diferentes orígenes. Esto facilita el modelado y el análisis de datos. Para crear una relación, simplemente seleccione las columnas que desea relacionar y seleccione el tipo de relación. A esto, se podría añadir lo siguiente:

- **Información sobre los orígenes de datos admitidos**: Power BI Desktop admite una amplia variedad de orígenes de datos, incluidos archivos locales, bases de datos, servicios en la nube y aplicaciones SaaS. Algunos de los orígenes de datos admitidos son:
  - **Bases de datos relacionales**: Microsoft SQL Server, Oracle, MySQL, PostgreSQL, etc.
  - **Bases de datos no relacionales**: Azure Cosmos DB, Amazon Redshift, Google BigQuery, etc.
  - **Hojas de cálculo**: Excel, Google Sheets, etc.
  - **Archivos de texto**: CSV, JSON, XML, etc.
  - **Servicios en la nube**: Azure Data Lake Storage, Amazon S3, Google Cloud Storage, etc.
- **Información sobre las funciones de limpieza y formato**: Power BI Desktop ofrece una amplia gama de funciones para limpiar y formatear datos. Algunas de estas funciones incluyen:
  - **Dividir columnas**: dividir una columna en varias columnas en función de un criterio.
  - **Cambiar el nombre de columnas**: cambiar el nombre de una columna.
  - **Cambiar los tipos de datos**: cambiar el tipo de datos de una columna.
  - **Trabajar con fechas**: formatear fechas, calcular intervalos de fechas, etc.

- **Información sobre las relaciones entre columnas**: Las relaciones entre columnas permiten a Power BI Desktop unir datos de diferentes orígenes. Hay dos tipos de relaciones:
    - **Relaciones 1:1**: una fila en una tabla se relaciona con una sola fila en otra tabla.
    - **Relaciones 1: N**: una fila en una tabla se puede relacionar con varias filas en otra tabla.

Ejemplos de uso

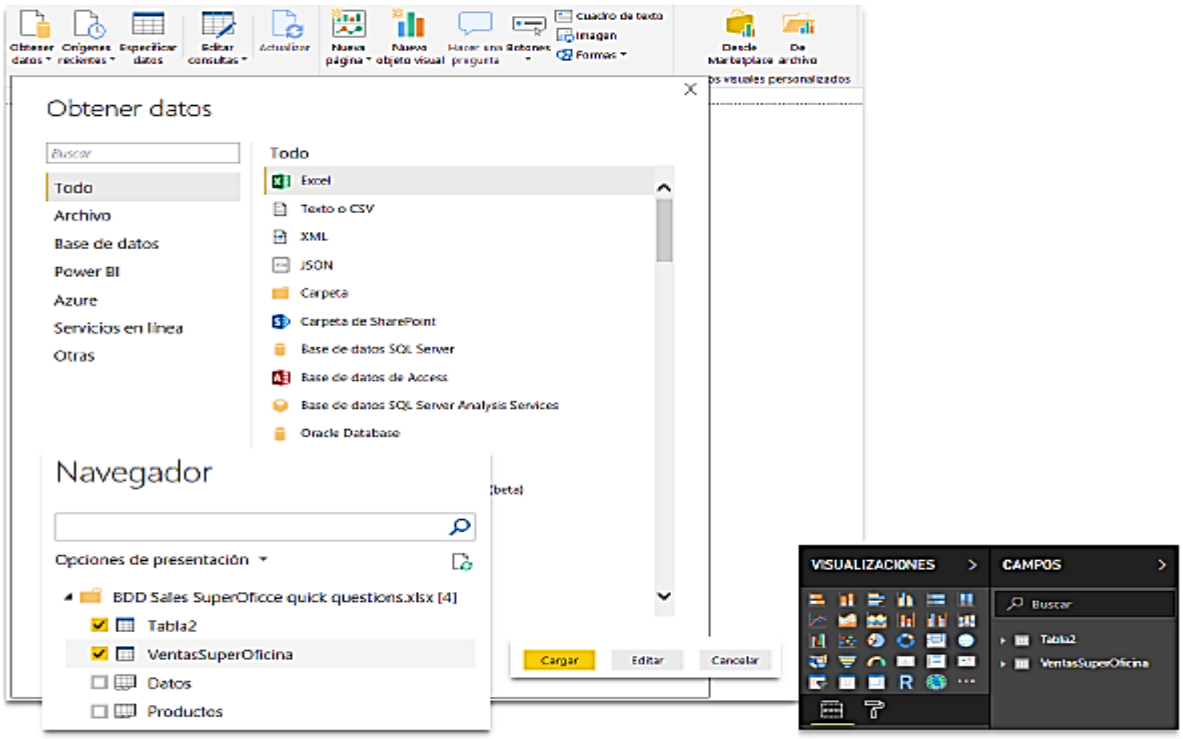
Power BI Desktop se puede utilizar para una amplia variedad de tareas de análisis de datos, como:

- **Análisis de ventas**: Power BI Desktop se puede utilizar para analizar datos de ventas para identificar tendencias, patrones y oportunidades.
- **Análisis de marketing**: Power BI Desktop se puede utilizar para analizar datos de marketing para medir la eficacia de las campañas y optimizar las estrategias.
- **Análisis financiero**: Power BI Desktop se puede utilizar para analizar datos financieros para tomar decisiones informadas.

En Power BI Desktop, seleccione Obtener datos. Aparecerá la colección de orígenes de datos para que pueda elegir uno de ellos independientemente del origen de datos que elija, Power BI se conecta a él y le muestra los datos disponibles como en la siguiente Figura 15:

Figura 15

Ventana Obtener datos, de Power BI



Fuente: (Quantec, s.f.)

Después de conectarse, la primera ventana que verá será Navegador. En ella aparecen las tablas o entidades del origen de datos. Si hace clic en una de ellas, obtendrá una vista previa de sus contenidos. Luego, puede importar las tablas o entidades seleccionadas inmediatamente, o seleccionar Editar para transformar y limpiar los datos antes de importarlos.

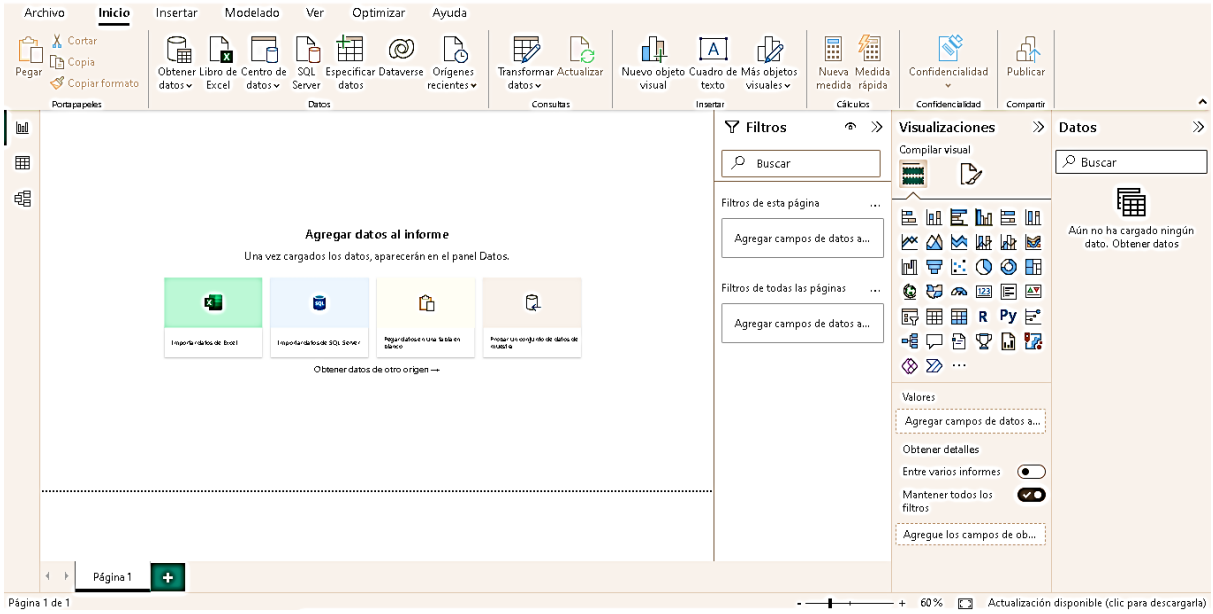
En la vista Informe de Power BI Desktop, se puede empezar a generar informes.

La vista Informe tiene cinco áreas principales:

1. La cinta de opciones, que muestra las tareas comunes asociadas con los informes y visualizaciones
2. La vista Informe o lienzo (Figura 16), donde se crean y se organizan las visualizaciones

Figura 16

Vista Informe, en Power BI



Fuente: Power BI

3. El área de pestaña Páginas situada a lo largo de la parte inferior, que permite seleccionar o agregar una página de informe
4. El panel Visualizaciones, donde puede cambiar las visualizaciones, personalizar los colores o ejes, aplicar filtros, arrastrar campos, etc.
5. El panel de Campos, donde elementos de consulta y filtros se pueden arrastrar a la vista Informe, o bien al área Filtros del panel Visualizaciones.

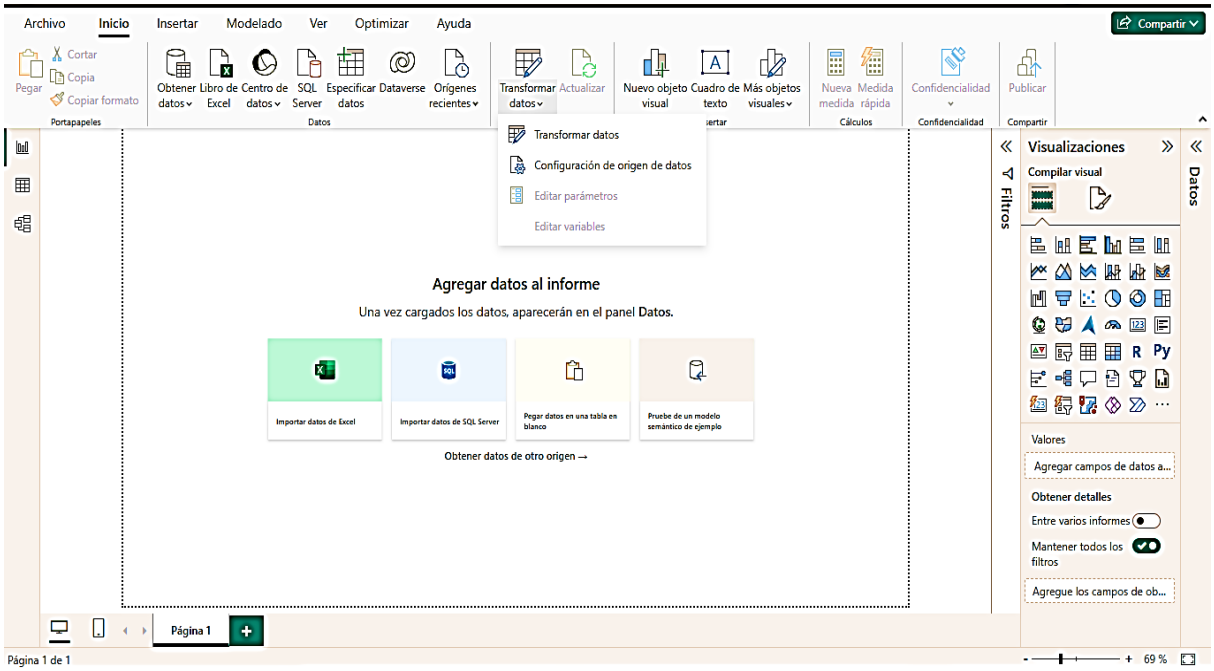
Limpieza y transformación de datos

Power BI Desktop incluye Transformar datos o su equivalente Editor de consultas en otras versiones de Power BI, una herramienta eficaz para dar forma a los datos y transformarlos con el objetivo de que estén listos para sus modelos y visualizaciones. Al seleccionar Transformar datos en la ventana Navegador, se inicia el Editor de consultas y se rellena con las tablas u otras entidades que haya seleccionado en el origen de datos.

También se puede iniciar el Transformar datos (Figura 17) directamente desde Power BI Desktop, utilizando el botón Transformar datos situado en la cinta Inicio.

Figura 17

Transformar datos en Power BI



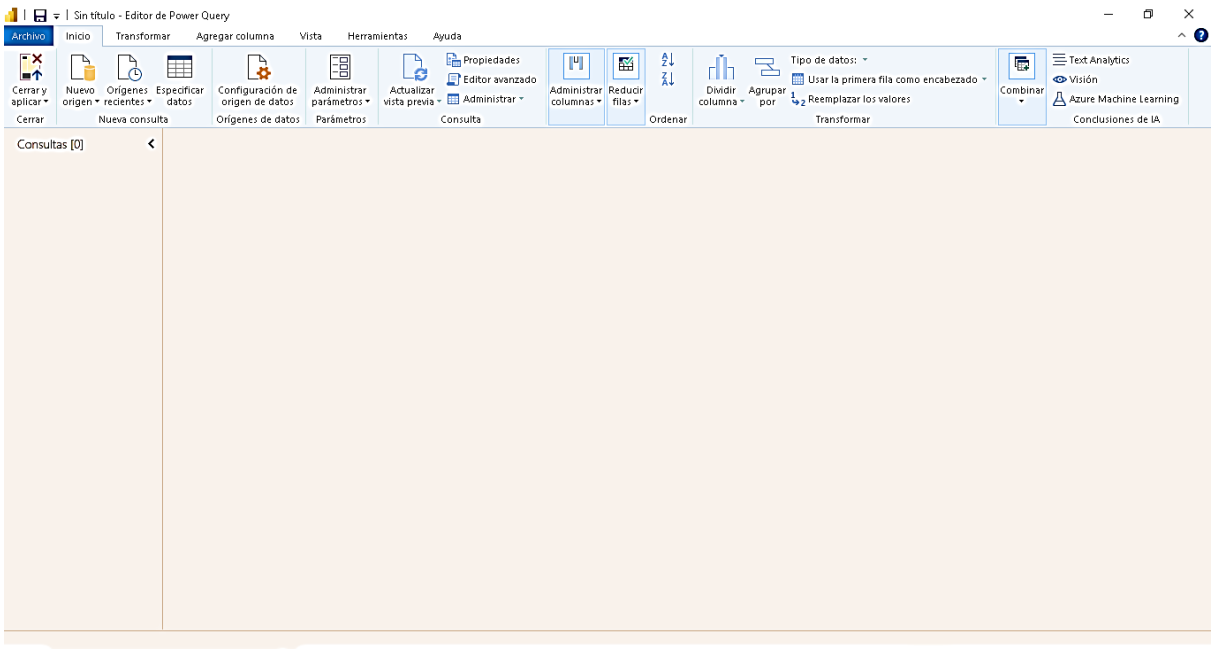
Fuente: Power BI

Una vez cargado el Editor de consultas (Figura 18) con datos listos para que les dé forma, verá una serie de secciones:

1. En la cinta de opciones aparecen muchos botones activos que permiten interactuar con los datos de la consulta.
2. En el panel izquierdo se enumeran las consultas (una por cada tabla o entidad), disponibles para su selección, visualización y cambio de forma.
3. En el panel central se muestran los datos de la consulta seleccionada, disponibles para darles forma.
4. Aparece la ventana Configuración de consulta, con una lista de las propiedades de la consulta y de los pasos aplicados.

Figura 18

Editor de Power Query



Fuente: Power BI

Para dar inicio a la creación de un cuadro de mando, primero debemos tener una base de datos que servirá de base para desarrollar un modelo de datos que de sentido a toda nuestra información.

El proceso de crear un modelo de datos implica muchas veces la transformación y carga de datos para poder armar las visualizaciones que permitirán analizar el negocio y dar respuestas a sus objetivos.

El ciclo de un modelo de datos que utiliza la inteligencia de negocios es el siguiente:

1. **Base de Datos**
    - Visualización
    - Gráficas
    - Matrix
    - Segmentación

2. DAX

- Columnas Calculadas
- Campos Calculados
- Transformación de Datos
- Power Query

3. Mejoramiento del modelo

- Relaciones
- Tablas
- Columnas
- Jerarquías

El modelo de análisis aplicando la Inteligencia de Negocios BI, toma mayor relevancia cuando se enriquece con DAX, que permite utilizar campos calculados y medidas.

CAPÍTULO DOS

LENGUAJE DAX

Introducción

Según (Quantec, s.f.):

DAX; sus siglas representan a Expresiones de Análisis de Datos, es un lenguaje de fórmulas desarrollado por Microsoft para Power BI. Se utiliza para realizar cálculos personalizados en modelos de datos tabulares y está compuesto por una cantidad de funciones que se pueden combinar para generar fórmulas y expresiones.

Su funcionamiento es similar a trabajar en Excel lo que hace de su sintaxis simple y fácil de utilizar para realizar un cálculo.

Inclusive ha heredado algunas funciones propias de Excel con el mismo nombre y funcionalidad.

¿Qué es DAX?

DAX es una colección de funciones, operadores y constantes que se pueden usar en una fórmula o expresión, para calcular y devolver uno o más valores. DAX le ayuda a crear información nueva a partir de datos ya existentes en el modelo.

¿Por qué es tan importante DAX?

Crear un archivo de Power BI Desktop nuevo e importar algunos datos en él es fácil. Incluso puede crear informes que muestren información valiosa sin usar las fórmulas DAX en absoluto. Pero, ¿qué ocurre si necesita analizar el porcentaje de crecimiento por categorías de producto y para intervalos de fechas diferentes? ¿O bien, si debe calcular el crecimiento interanual en comparación con las tendencias del mercado? Las fórmulas DAX proporcionan esta y muchas otras capacidades igual de importantes. Obtener información sobre cómo crear fórmulas DAX eficaces le ayudará a aprovechar los datos al máximo. Cuando obtiene la

información que necesita, puede empezar a resolver problemas empresariales reales que afectan los resultados finales.

Requisitos previos

Es posible que ya esté familiarizado con la creación de fórmulas en Microsoft Excel y ese conocimiento será útil para comprender DAX. Pero incluso si no tiene ninguna experiencia con las fórmulas de Excel, los conceptos descritos aquí le ayudarán a empezar a crear fórmulas de DAX y solucionar problemas de inteligencia empresarial del mundo real inmediatamente.

Nos centraremos en comprender las fórmulas DAX que se usan en los cálculos, más específicamente, en columnas calculadas y medidas. Debe estar familiarizado con el uso de Power BI Desktop para importar datos y agregar campos a un informe. También debe estar familiarizado con los conceptos fundamentales de medidas y columnas calculadas.

Contexto

El contexto es uno de los conceptos de DAX más importantes. Hay dos tipos de contexto en DAX: contexto de fila y contexto de filtro. En primer lugar, examinaremos el contexto de fila.

Contexto de fila

Es más fácil pensar en el contexto de fila como la fila actual. Se aplica siempre que una fórmula tiene una función use filtros para identificar una fila individual en una tabla. De manera inherente, la función aplicará un contexto de fila para cada fila de la tabla que se está filtrando. Este tipo de contexto de fila se aplica con mayor frecuencia en medidas.

Contexto de filtro

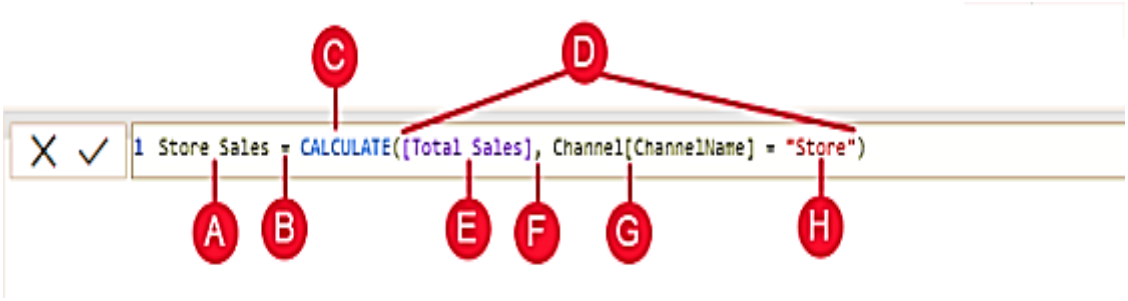
El contexto de filtro es un poco más difícil de entender que el contexto de fila. Para simplificarlo, piense en el contexto de filtro como uno o varios filtros aplicados en un cálculo que determina un resultado o valor.

El contexto de filtro no existe en lugar del contexto de fila; más bien, se aplica, además del contexto de fila. Por ejemplo, para restringir aún más los valores que quiera incluir en un

cálculo, puede aplicar un contexto de filtro que no solo especifique el contexto de fila, sino que también especifique un valor determinado (filtro) en ese contexto de fila.

El contexto de filtro se puede identificar fácilmente en los informes. Por ejemplo, cuando agrega TotalCost a una visualización y, a continuación, Year y Region, define un contexto de filtro que selecciona un subconjunto de datos basándose en un año y región determinados.

¿Por qué es tan importante para DAX el contexto de filtro? Ha visto que el contexto de filtro se puede aplicar agregando campos a una visualización. El contexto de filtro también se puede aplicar en una fórmula de DAX mediante la definición de un filtro con funciones como ALL, RELATED, FILTER o CALCULATE, mediante relaciones, así como mediante otras medidas y columnas. Por ejemplo, echemos un vistazo a la siguiente fórmula en una medida denominada Store Sales:



Para entender mejor esta fórmula, podemos desglosarla, similar a como podemos hacerlo otras fórmulas.

Esta fórmula incluye los siguientes elementos de sintaxis:

A. El nombre de medida, Store Sales.

B. El operador signo igual (=) que indica el principio de la fórmula.

C. La función CALCULATE, que evalúa una expresión, como argumento, en un contexto modificado por los filtros especificados.

D. Los paréntesis () que envuelven una expresión que contiene uno o más argumentos.

E. Una medida [Total Sales] en la misma tabla como una expresión. La medida Total Sales tiene la fórmula: =SUM(Sales[SalesAmount]).

F. Una coma (,), que separa el primer argumento de expresión del argumento de filtro.

G. La columna completa a la que se hace referencia, Channel[ChannelName]. Se trata de nuestro contexto de fila. Cada fila de esta columna especifica un canal como Store, Online, etc.

H. El valor concreto Store como filtro. Este es el contexto de filtro

Esta fórmula garantiza que únicamente se calculen los valores de ventas definidos por la medida Total Sales y solamente para las filas de la columna Channel[ChannelName] con el valor Store utilizado como filtro.

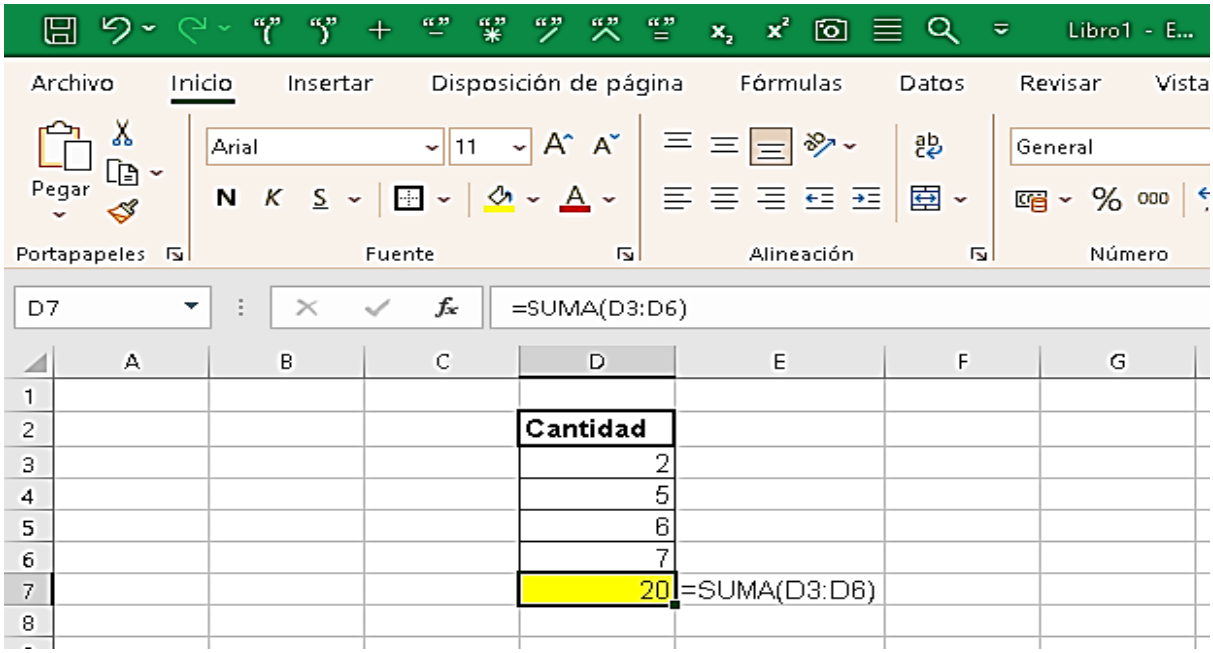
Como puede imaginar, la posibilidad de definir el contexto de filtro dentro de una fórmula ofrece grandes y potentes capacidades. La capacidad de hacer referencia solamente a un valor determinado en una tabla relacionada es solo un ejemplo. No se preocupe si no comprende por completo el concepto de contexto inmediatamente. A medida que cree sus propias fórmulas, comprenderá mejor el contexto y por qué es tan importante en DAX.

Sintaxis de DAX

La sintaxis de DAX es similar a la de Excel, lo que hace que sea fácil de aprender para los usuarios que ya están familiarizados con ese lenguaje.

Las fórmulas en Excel están orientadas a realizar cálculos basados en celdas y rangos de éstas; por ejemplo, en la celda A7 podemos escribir la siguiente expresión:

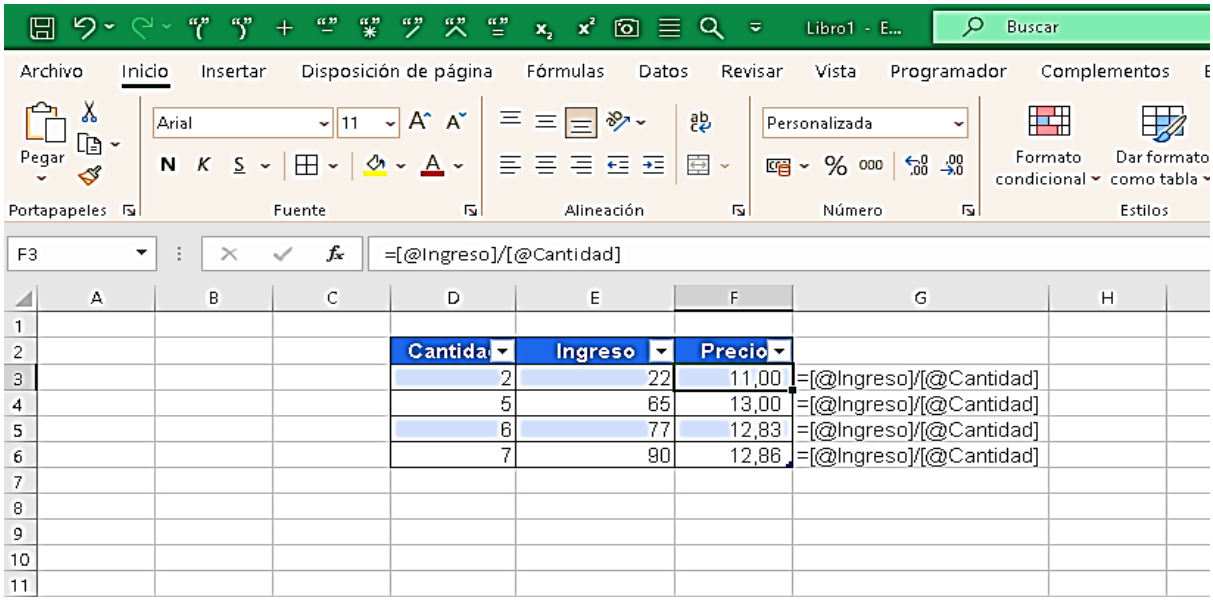
=SUMA(D3:D6) cierto, esto es lo tradicional que ya conocemos de Excel.



También se tiene una nueva sintaxis, que nos ofrece las nuevas versiones de Excel conocidas como **tabla**, al trabajar con un conjunto de celdas, y esta nos permite hacer referencia a las columnas de la tabla con una expresión diferente.

Ejemplo:

"=[@Ingresos]/[@Cantidad]", las tablas permiten usar la sintaxis tanto en celdas como en columnas.



En el ejemplo, basta con establecer en una de las celdas del campo Precio y al dar enter, la misma fórmula, tal cual como se la escribió originalmente se repite en las demás celdas.

Pero a priori a hacer todo esto, se da formato de tabla a la base de datos que se tenga, caso contrario no se podrá realizar fórmulas como la indicada.

Según (Microsoft, 2023), a diferencia de Excel, DAX solo funciona en tablas. Todas las fórmulas deben hacer referencia a columnas dentro de tablas. Por ejemplo, en DAX escribimos la multiplicación anterior de esta manera:

Precio [NombreTabla] = Ingreso [NombreTabla] / Cantidad [NombreTabla]

Como se puede ver, cada columna tiene como prefijo el nombre de su tabla. En Excel, no proporcionamos el nombre de la tabla porque las fórmulas de Excel funcionan dentro de una sola tabla. Sin embargo, DAX funciona en un modelo de datos que contiene muchas tablas. Como consecuencia, debemos especificar el nombre de la tabla porque dos columnas en tablas diferentes pueden tener el mismo nombre.

Resumiendo: Características generales del lenguaje DAX:

- Una gran parte de las funciones de DAX se denominan igual que las de Excel y tienen el mismo comportamiento, solo cambia la referencia, así en el caso de DAX es una columna o tabla de referencia mientras que en Excel son una celda o rango de celdas.
- Con DAX podemos crear columnas, campos (medidas) y tablas.
- DAX cuenta con una tipología de funciones propias que permite devolver una tabla de valores o realizar cálculos sobre una tabla de valores como entrada y que no está disponible en Excel.

Principales diferencias entre el lenguaje DAX y las fórmulas de Excel son:

- En DAX no existe el concepto celda de Excel. DAX trabaja sobre tablas, las funciones esperan tablas o columnas
- Una función de DAX siempre hace referencia a una columna o una tabla. Si solo desea usar valores concretos de una tabla o columna, puede agregar filtros a la fórmula.
- DAX aporta nuevas funciones que fortalecen el potencial de Excel como son: Funciones de inteligencia de tiempo e Iteradores.

Variables

En DAX (Escuela De Postgrado Y Educación Continua, Facultad De Ciencias Físicas y Matemáticas, Universidad de Chile, s.f.), se pueden declarar variables dentro de una fórmula por múltiples motivos. En primer lugar, mejora la legibilidad y el mantenimiento de las fórmulas, debido a que se evita la repetición de código. En segundo lugar, mejora el rendimiento ya que las variables se evalúan una vez. Por último, permite realizar pruebas sencillas de una fórmula compleja en el momento del su diseño. Por ejemplo, para calcular la rentabilidad año a año se puede usar la siguiente medida sin usar variables.

Profit YoY % = DIVIDE([Profit] - CALCULATE([Profit], SAMEPERIODLASTYEAR('Fact Internet Sales'[OrderDate])), CALCULATE([Profit], SAMEPERIODLASTYEAR('Fact Internet Sales'[OrderDate])))

La definición de la medida Profit YoY% se puede reescribir con el uso de una variable que guarde la utilidad del año anterior.

Profit Year % = VAR ProfitPriorYear = CALCULATE([Profit], SAMEPERIODLASTYEAR('Fact Internet Sales'[OrderDate]))

RETURN DIVIDE([Profit]-ProfitPriorYear, ProfitPriorYear)

Muchas funciones en DAX funcionan de la misma manera que la función equivalente de Excel. Por ejemplo, el SI La función se lee de la misma manera en DAX y en Excel.

Por ejemplo:

Excel: SI ([@SalesAmount]> 10, 1, 0)

DAX: SI (Ventas [SalesAmount]> 10, 1, 0)

Con referencia a estas dos sintaxis, se facilita el camino a la sintaxis de DAX, que es muy similar a la sintaxis de las tablas de Excel, aunque con algunas diferencias:

- DAX solo hace referencia a las columnas de las tablas, no existe el concepto de celdas.
- Se aplica a las columnas calculadas y campos calculados o medidas.
- Cuando se aplica a un campo calculado, la sintaxis incluye el nombre del nuevo campo.

Tablas calculadas

Con DAX se pueden escribir fórmulas para agregar una tabla calculada al modelo de datos.

La fórmula puede duplicar o transformar los datos del modelo existente, o crear una serie de

datos para crear una tabla.

Los datos de las tablas calculadas siempre se importan en el modelo, por lo que aumenta el tamaño de almacenamiento del modelo y puede prolongar la duración de la actualización de los datos. Las tablas calculadas pueden ser útiles en diversos escenarios, como para crear tablas de fechas, dimensiones realizadoras de roles o análisis de hipótesis.

Tablas de fechas

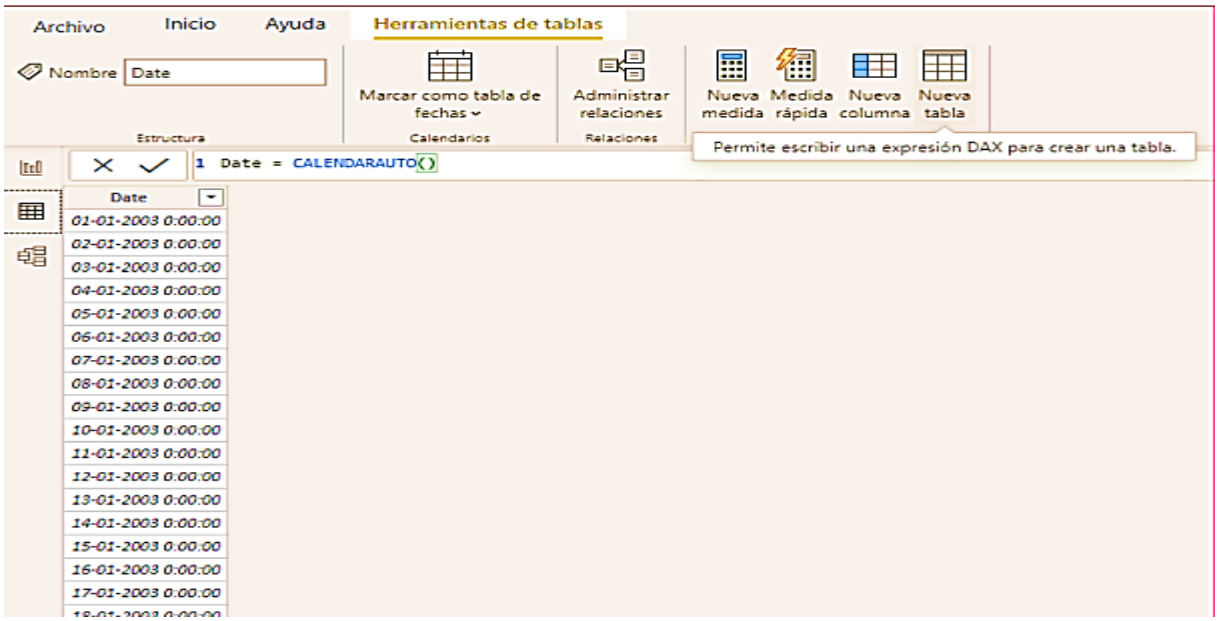
Las tablas de fechas son necesarias para aplicar filtros de hora especiales conocidos como inteligencia de tiempo. Una familia de funciones de DAX, conocida como funciones de inteligencia de tiempo, solo funciona correctamente cuando se configura una tabla de fechas. Cuando los datos originales no incluyen una tabla de fechas esta se puede crear como una tabla calculada a través de las funciones CALEANDAR o CALEANDARAUTO de DAX.

Para crear una nueva tabla se debe ir a Herramientas de tablas en el modelo de datos y dar clic Nueva tabla, tras esto se habilita un cuadro de texto dentro del cual se pueden escribir fórmulas en DAX para generar una nueva tabla.

Por ejemplo, en la Figura 19 se muestra como generar una tabla calculada con la función CALENDARAUTO, la cual devuelve una tabla con una sola columna denominada "Date" que contiene un conjunto contiguo de fechas entre la fecha mínima y máxima de los datos.

Figura 19

Tabla calculada con la función CALENDARAUTO()



Fuente: (Escuela De Postgrado Y Educación Continua, Facultad De Ciencias Físicas y Matemáticas, Universidad de Chile, s.f.)

Dimensiones realizadoras de roles

Cuando dos tablas de modelos tienen varias relaciones, podría deberse a que el modelo tiene

una dimensión realizadora de roles.

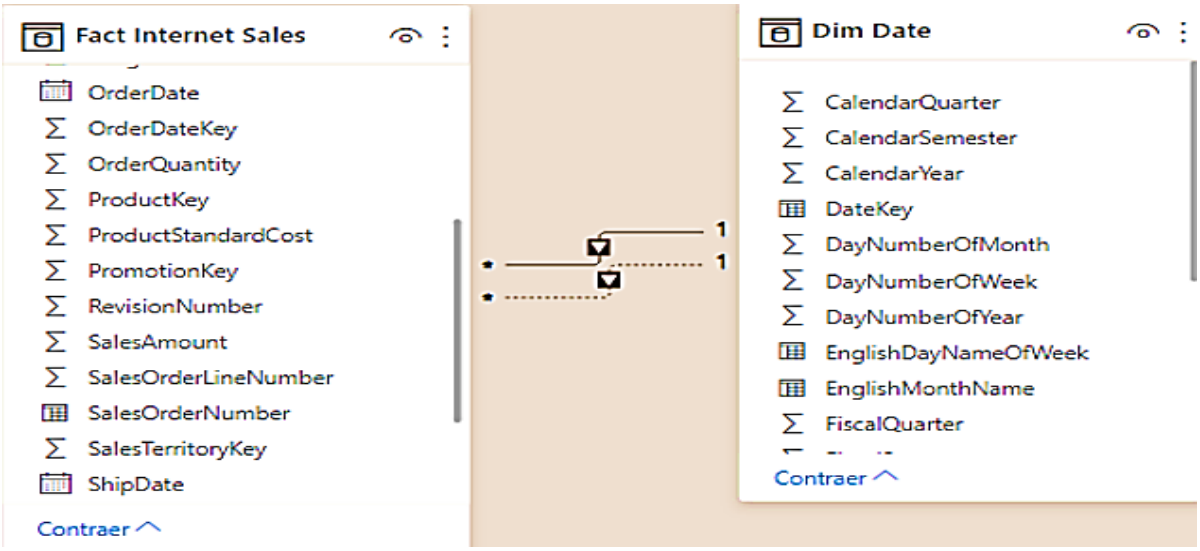
Por ejemplo, se tiene la tabla Fact Internet Sales que incluye dos columnas de fecha, OrderDateKey y ShipDateKey, ambas columnas están relacionadas con la columna DateKey de la tabla Dim Date. Esto es ilustrado en la Figura 20.

En este caso, la tabla Dim Date se describe como una dimensión realizadora de roles, ya que

podría desempeñar el rol de fecha de pedido o fecha de envío.

Figura 20

Relación activa e inactiva entre la tabla Fact Internet Sales y Dim Date. La relación activa relaciona el campo



Fuente: (Escuela De Postgrado Y Educación Continua, Facultad De Ciencias Físicas y Matemáticas, Universidad de Chile, s.f.)

Los modelos de datos solo permiten una relación activa entre las tablas, lo que es indicado a través de una línea sólida. De forma predeterminada, Power BI utiliza la relación activa para propagar los filtros, en este caso, la relación entre DateKey y OrderDateKey. En este caso, un mejor diseño del modelo podría ser tener dos tablas de fecha, cada una con una relación activa con la tabla Fact Internet Sales. En la Figura 21 se muestra cómo crear esta tabla y en la Figura 22 el modelo de datos resultante.

Figura 21

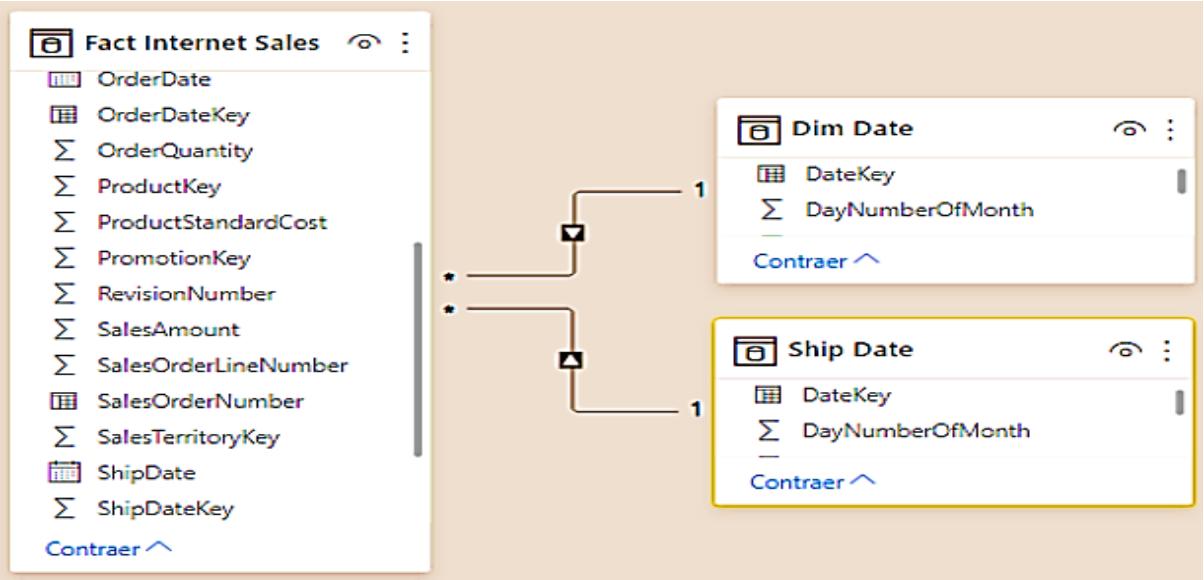
Duplicar tabla Dim Date bajo el nombre Ship Date



Fuente: (Escuela De Postgrado Y Educación Continua, Facultad De Ciencias Físicas y Matemáticas, Universidad de Chile, s.f.)

Figura 22

La tabla Fact Internet Sales está relacionada a través de sus campos OrderDateKey y ShipDateKey con el campo DateKey de la tabla Dim Date y Ship Date respectivamente



Fuente: (Escuela De Postgrado Y Educación Continua, Facultad De Ciencias Físicas y Matemáticas, Universidad de Chile, s.f.)

Análisis de hipótesis

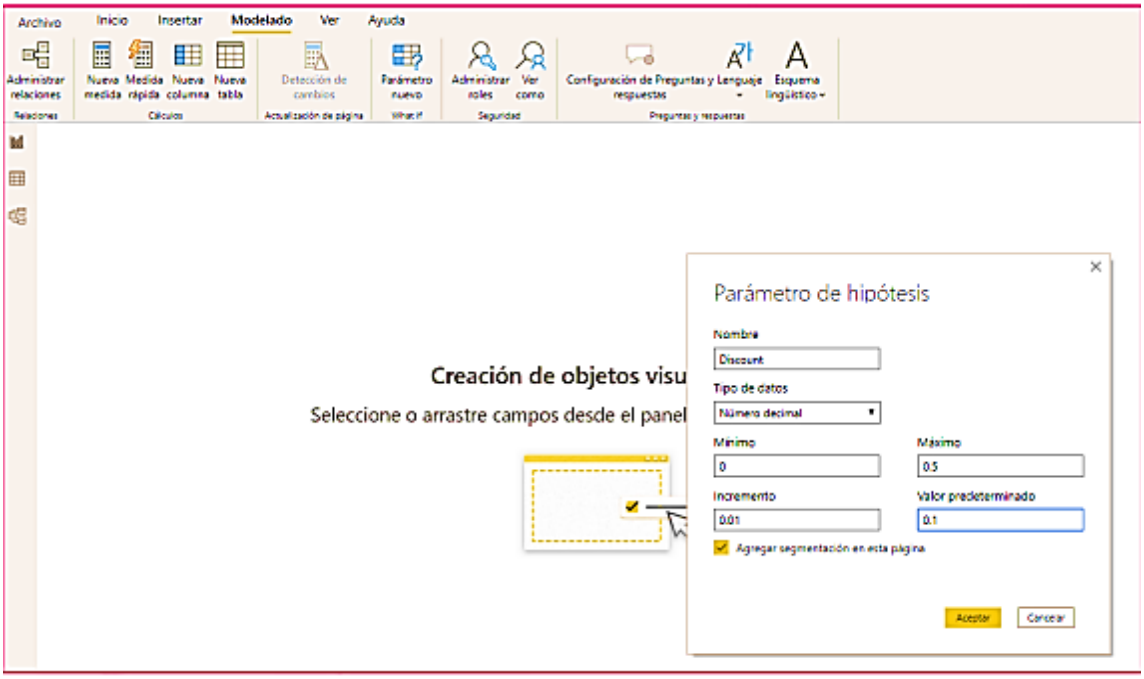
Power BI Desktop admite una característica llamada parámetros de hipótesis o parámetro what-if. Un parámetro what-if permite segmentar los datos y agregarlos a una tabla calculada al modelo. Por ejemplo, podría crear un informe que permita al personal de ventas ver sus compensaciones si satisfacen determinados objetivos o porcentajes de venta, o ver la influencia que tiene el incremento en las ventas sobre unos mayores descuentos.

En particular, estas tablas calculadas no están relacionadas con otras tablas de modelos porque no se usan para propagar los filtros.

Para crear una tabla what-if se debe estar en la vista Informe, en el menú Modelado y dar click a Parámetro nuevo, tras esto se abrirá una ventana en la que se puede definir el patrón de la columna que tendrá la tabla. En la Figura 23 se muestra cómo crear el parámetro de hipótesis Discount como una columna que toma valores en el intervalo [0, 0.5] con saltos de 0.01. En la Figura 24 se muestra la tabla creada en el modelo de datos a partir del parámetro de hipótesis Discount.

Figura 23

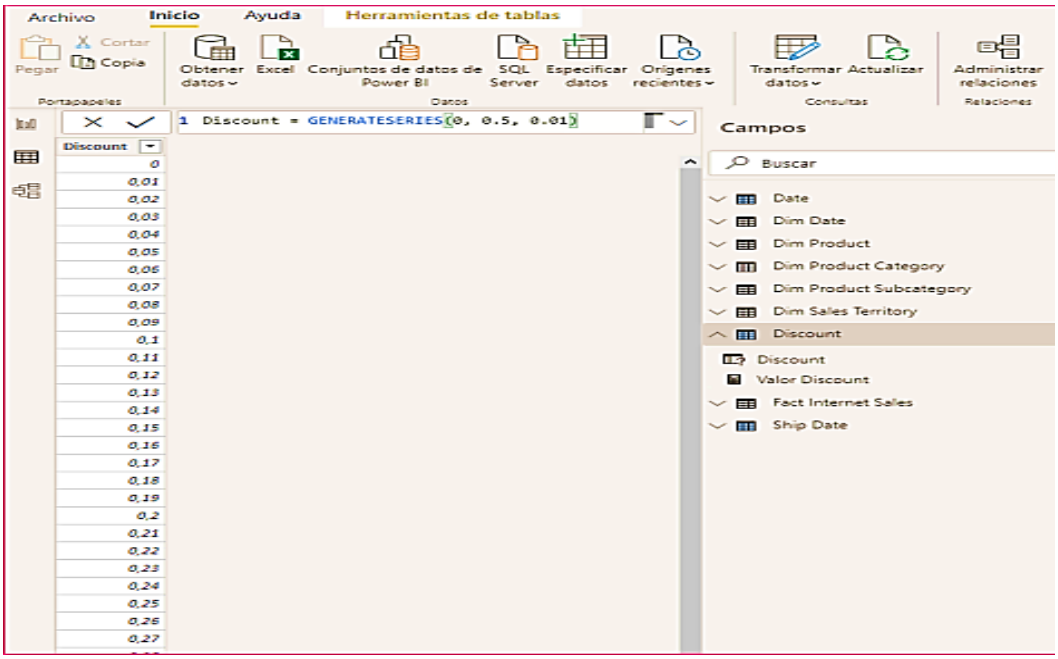
Menú de creación de un parámetro de hipótesis



Fuente: (Escuela De Postgrado Y Educación Continua, Facultad De Ciencias Físicas y Matemáticas, Universidad de Chile, s.f.)

Figura 24

Tabla calculada a partir de un parámetro de hipótesis



Fuente: (Escuela De Postgrado Y Educación Continua, Facultad De Ciencias Físicas y Matemáticas, Universidad de Chile, s.f.)

Columnas calculadas

Las columnas calculadas son columnas adicionales que se agregan a una tabla o consulta existente de Power BI. Se crean utilizando expresiones DAX, que permiten realizar cálculos personalizados sobre los datos de la tabla.

Creación de columnas calculadas

Para crear una columna calculada, siga estos pasos:

1. En el panel de navegación, haga clic con el botón derecho en la tabla o consulta en la que desea agregar la columna calculada.
2. Seleccione Agregar columna calculada.
3. En la barra de fórmulas, escriba la expresión DAX que desea utilizar para calcular los valores de la columna.
4. Presione Enter.

Expresión de la columna calculada

La expresión de la columna calculada debe devolver un valor para cada fila de la tabla. La expresión puede utilizar funciones DAX, columnas existentes o valores constantes.

Actualización de columnas calculadas

Cuando se actualizan los datos de origen, las columnas calculadas se actualizan automáticamente.

Usos de las columnas calculadas

Las columnas calculadas se pueden utilizar para una variedad de propósitos, como:

- Agregar nuevos datos a una tabla
- Realizar cálculos personalizados sobre los datos de una tabla
- Crear medidas
- Realizar segmentaciones
- Diferencias entre columnas calculadas y medidas

Las columnas calculadas y las medidas son dos tipos diferentes de objetos DAX que se utilizan para realizar cálculos personalizados. La principal diferencia entre ambos es que las columnas calculadas se almacenan en la tabla, mientras que las medidas se almacenan en el modelo.

Las columnas calculadas son una herramienta poderosa que se puede utilizar para agregar nuevos datos, realizar cálculos personalizados y crear medidas. Son una parte esencial de la creación de informes de Power BI.

Además de las explicaciones proporcionadas en el texto original, se pueden agregar los siguientes detalles:

Ejemplos de columnas calculadas:

- Una columna calculada que calcula el margen de beneficio de un producto.
- Una columna calculada que calcula la edad de un cliente.
- Una columna calculada que calcula el número de veces que un cliente ha visitado un sitio web.

Recomendaciones para la creación de columnas calculadas:

- Utilice nombres descriptivos para las columnas calculadas.
- Documente las columnas calculadas para que sea fácil entender su propósito.
- Pruebe las columnas calculadas para asegurarse de que funcionan correctamente.

Usos avanzados de las columnas calculadas:

- Las columnas calculadas se pueden utilizar para crear jerarquías.
- Las columnas calculadas se pueden utilizar para crear filtros.
- Las columnas calculadas se pueden utilizar para crear tablas dinámicas.

Campos calculados o medidas

Los campos calculados, son los cálculos que se utilizan para el análisis de información. Se crean al escribir una expresión DAX, basada en los campos de una tabla y de una columna calculada.

Sus valores no se almacenan en disco, como sucede con las columnas calculadas, sino que el motor de Power BI los calcula al momento que se genera una visualización.

No son solamente fórmulas que se agregan a un panel para crear visualizaciones gráficas y en tablas, sino que ofrecen una potencia en sus cálculos y mayor flexibilidad de análisis.



Columnas calculadas vs campos calculados

Es importante saber cuándo debo escribir una columna calculada y cuando un campo calculado, aunque siempre va a depender de su aplicación.

Como aplicar cada caso:

- Utilizar columnas calculadas cuando se requieren cálculos que interactúen fila a fila.
- Utilizar campos calculados si necesita utilizar, filtros y segmentación de datos por filas o columnas.

Cálculos

Antes de trabajar en fórmulas más complejas, debes aprender los conceptos básicos de DAX. Esto incluye la sintaxis de DAX, los diferentes tipos de datos que DAX puede manejar, los operadores básicos y cómo hacer referencia a columnas y tablas.

Usamos DAX para calcular valores sobre columnas en tablas. Podemos sumar, calcular y buscar números, pero al final, todos los cálculos involucran tablas y columnas. Por lo tanto, la primera sintaxis que debe aprender es cómo hacer referencia a una columna en una tabla.

El formato general es escribir el nombre de la tabla entre comillas simples, seguido del nombre de la columna entre corchetes, de la siguiente manera: ´[Cantidad de ventas]

Se puede omitir las comillas simples si el nombre de la tabla no comienza con un número, no contiene espacios y no es una palabra reservada como fecha o suma.

El nombre de la tabla también es opcional, en caso de que hagamos referencia a una columna o medida dentro de la tabla donde definimos la fórmula. Por lo tanto, [Cantidad] es una referencia de columna válida, si está escrita en una columna calculada o en una medida. Aunque esta opción está disponible, se recomienda no omitir el nombre de la tabla.

Es de suma importancia poder distinguir entre medidas y columnas cuando se lee código DAX. El estándar de facto es utilizar siempre el nombre de la tabla en las referencias de columna y evitarlo siempre en las referencias de medidas, por ejemplo;

Ventas [Cantidad] * 2 - - Esto es una columna de referencia

[Importe de ventas] * 2 - - Esto es una referencia de medida

Aprenderá el fundamento de este estándar después de aprender sobre la transición de contexto.

Nota: El ejemplo de código anterior muestra comentarios en DAX por primera vez. DAX admite comentarios de una sola línea y comentarios de varias líneas. Los comentarios de una sola línea comienzan con -- o //, y la parte restante de la línea se considera un comentario.

= Ventas [Cantidad] * Ventas [Precio neto] -- Comentario de una sola línea

= Ventas [Cantidad] * Ventas [Costo unitario] // Otro ejemplo de comentario de una sola línea

Un comentario de varias líneas comienza con / * y termina con * /.

El analizador DAX ignora todo lo que se incluye entre estos marcadores y los considera un comentario.

= SI (Ventas [Cantidad]> 1, / * Primer ejemplo de un comentario de varias líneas

Cualquier cosa se puede escribir aquí y se ignora por DAX * / "Multi", / * Un caso de uso común de comentarios de varias líneas es comentar una parte del código existente

La siguiente instrucción IF se ignora porque se encuentra dentro de un comentario de varias líneas:

SI (Ventas [Cantidad] = 1, "Soltero", "Nota especial") * / "Soltero")

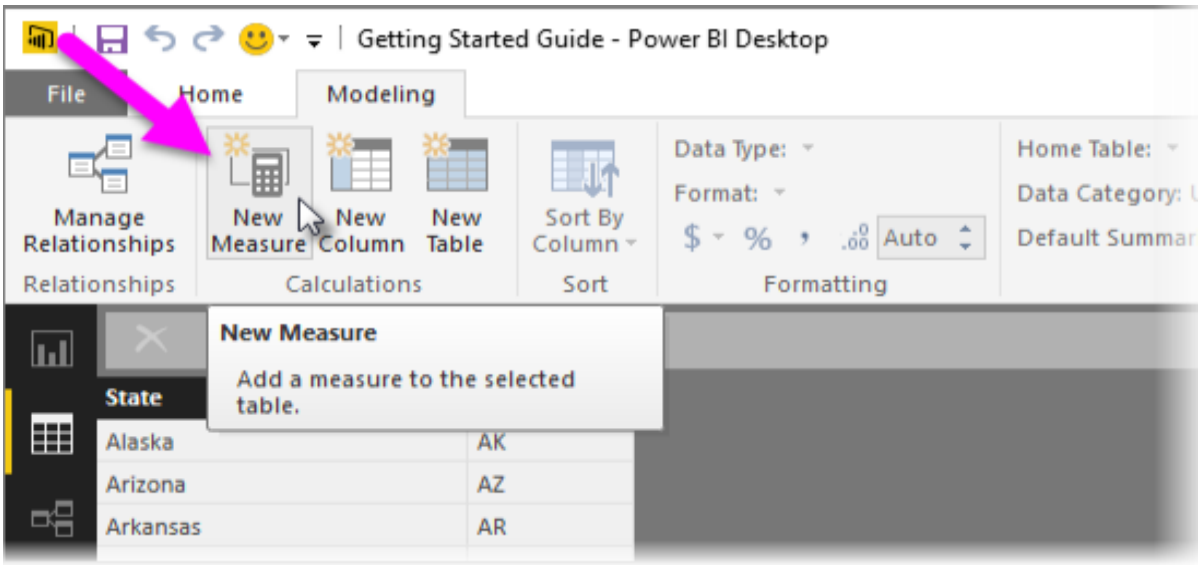
Es mejor evitar los comentarios al final de una expresión DAX en una medida, columna calculada o definición de tabla calculada. Es posible que estos comentarios no sean visibles al principio y que no sean compatibles con herramientas como DAX.

Creación de medidas calculadas

Use una medida calculada Figura 25 para calcular porcentajes o proporciones, o bien si necesita realizar agregaciones complejas. Para crear una medida usando una fórmula DAX, seleccione el botón Nueva medida desde la pestaña Modelado. Como antes, es mejor estar en la vista Datos de Power BI Desktop, ya que muestra la barra de fórmulas y facilita la escritura de la fórmula DAX.

Figura 25

Medidas calculadas

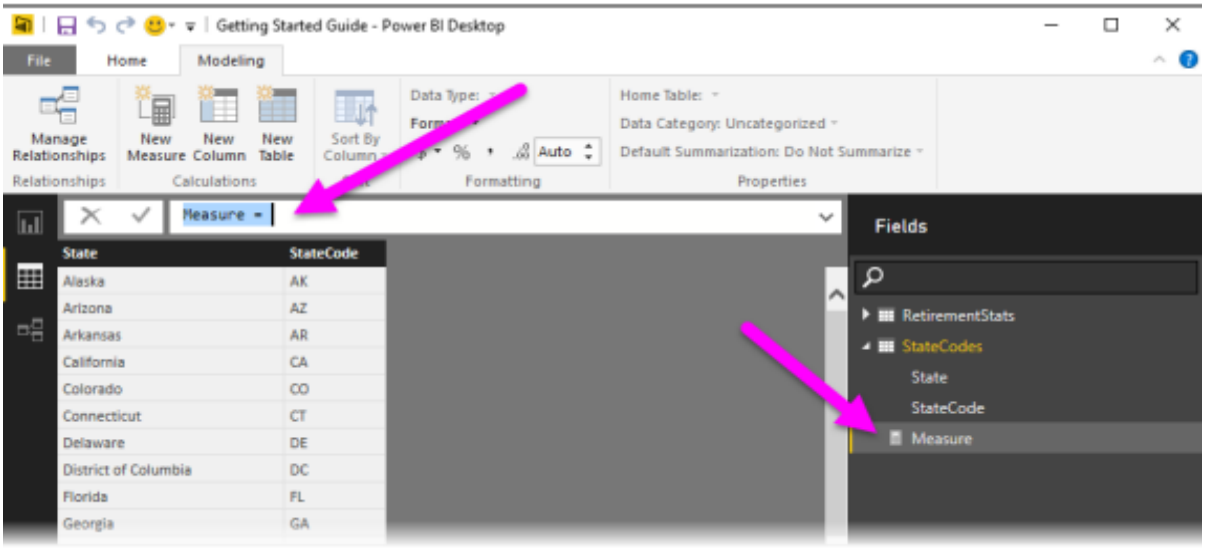


Fuente: (Quantec, s.f.)

Con medidas, aparece un nuevo icono de medida en el panel Campos con el nombre de la medida **Figura 26**. La barra de fórmulas se rellena otra vez con el nombre de la fórmula DAX (esta vez, con la medida).

Figura 26

Sintaxis de medida calculada



Fuente: (Quantec, s.f.)

Los elementos necesarios para una medida calculada son los mismos que para una columna calculada:

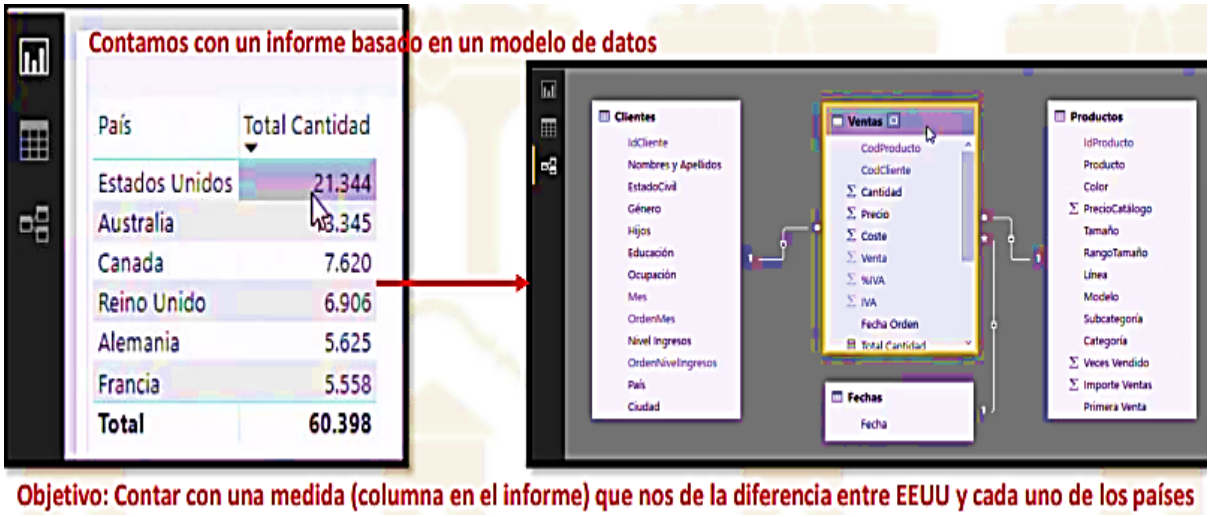
- un nuevo nombre de medida
- al menos una función o una expresión

Medidas rápidas en Power BI

Se dispone de un asistente tanto en Power BI como Power Pivot que nos facilita crear medidas rápidas generando el código automáticamente lo que al principio es de gran ayuda, pero es importante saber leer y comprender el código generado de forma automática.

Las medidas rápidas son expresiones de cálculo escritas en DAX que vienen integradas en Power BI y que son de gran utilidad en el proceso de aprendizaje de DAX ya que funcionan con un asistente intuitivo que nos permite programar la medida de forma visual sin usar el lenguaje, es decir, el asistente generara de forma automáticamente el código DAX.

Por ejemplo, contamos con un informe que contiene una sola columna que relaciona el total de ventas (cantidad) según país (fila), informe que está basado en un modelo de datos (activar vista de relaciones) compuesto por 4 tablas.

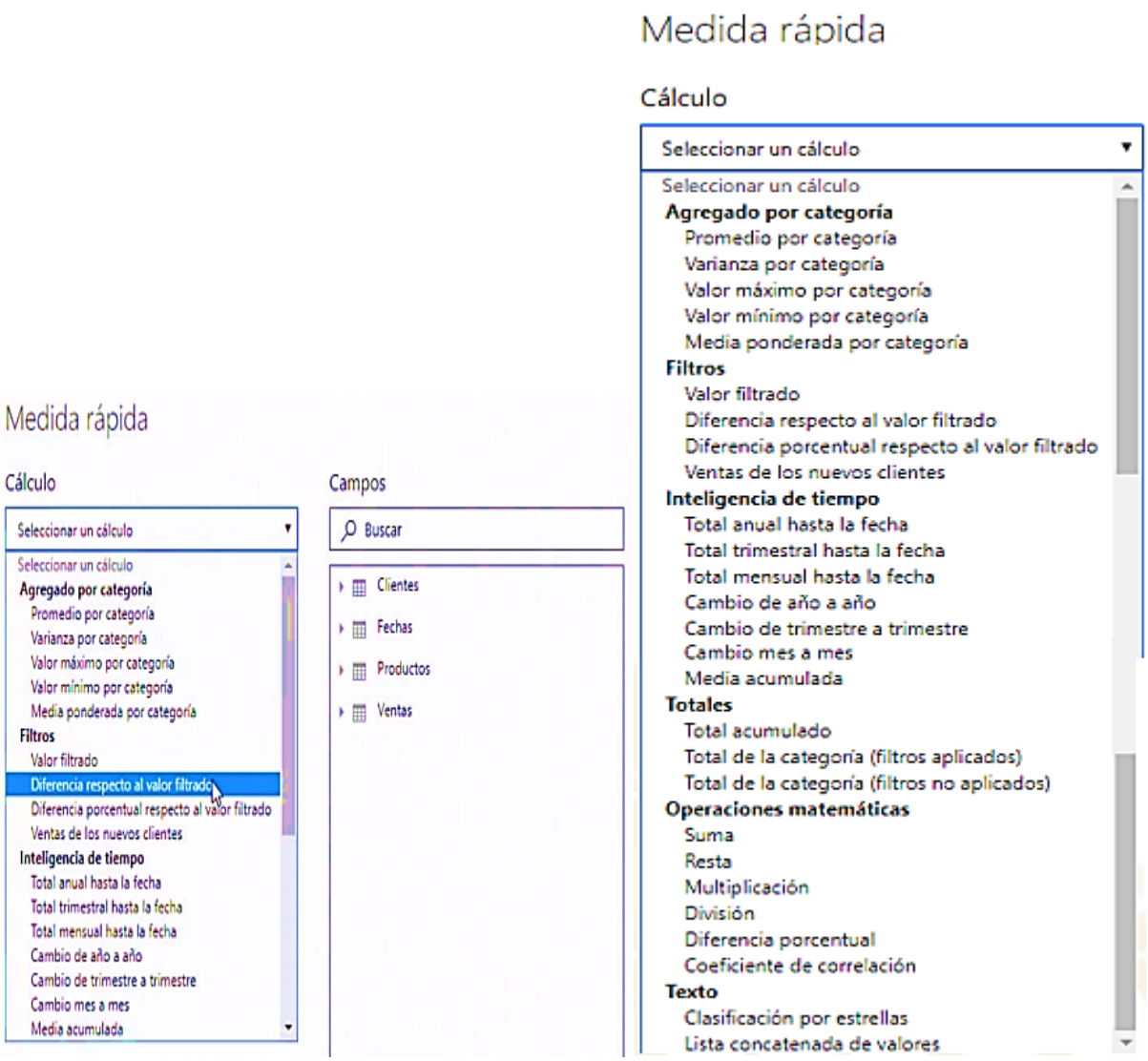


Nuestro objetivo es crear una medida que nos permita obtener una columna que muestre la diferencia entre la cantidad de EEUU y de cada uno de los países de la fila visibles. Esta medida la podemos crear usando el lenguaje DAX escribiendo las expresiones o bien a través de la opción de medidas rápida comentada y disponible en Power BI. Para acceder a la opción de medidas rápidas debemos acceder a la visualización informe o vista informe, panel de visualizaciones y en vista de campos vemos que en la propiedad valores tenemos los campos país y total cantidad, que son los campos disponibles y en el lateral derecho hay un pequeño triángulo y pulsando aparece un menú contextual que nos permite entre otras opciones la de crear una “Nueva medida rápida” y a través de la cual accedemos a un asistente para configurarla.



Una vez hemos accedido al asistente lo primero es seleccionar la opción cálculo para definir o seleccionar el tipo de cálculo deseado.

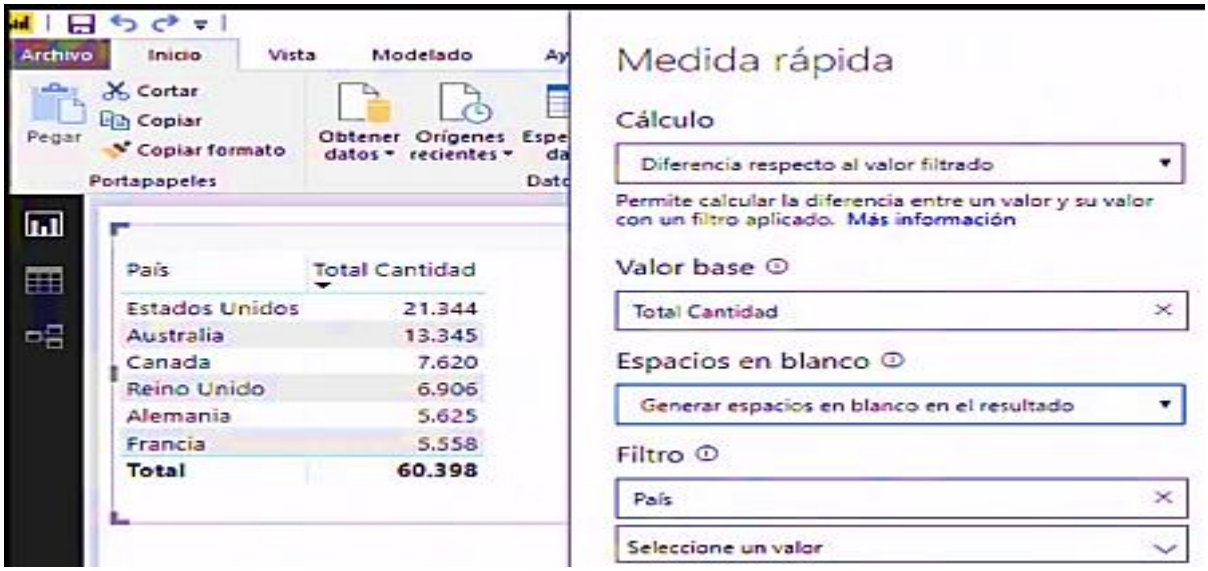
Destaca que nos pone a disposición un gran número de opciones de cálculo agrupadas o clasificadas en grupos o categorías en nuestro caso particular nos interesa ir al grupo filtros y seleccionar la opción “Diferencia respecto al valor filtrado” y de esta forma accedemos a configurarlo.



Por tanto, configuramos los valores con los parámetros básicos, tal y como se muestra en la ilustración, seleccionando el valor base como total cantidad qué es lo que estamos viendo en la tabla y es la medida que queremos comparar.

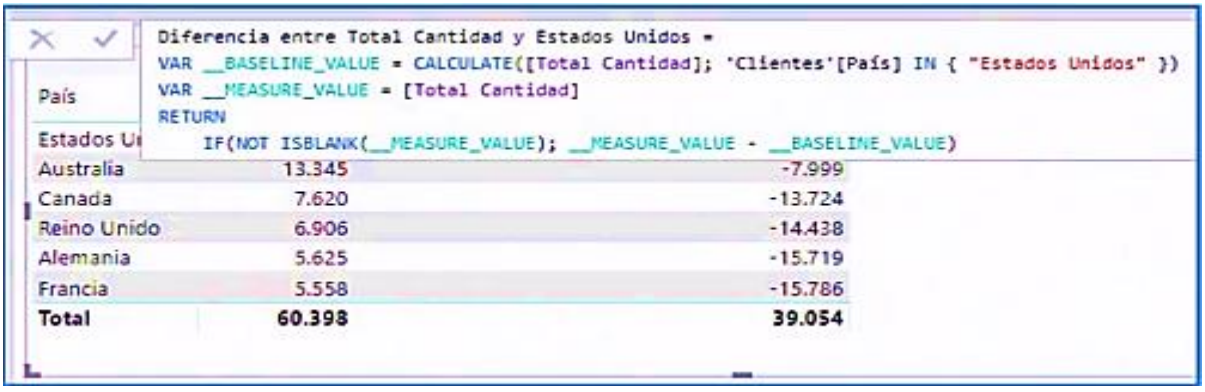
En el cuadro espacios en blanco podemos definir si lo que queremos es que se generan espacios en blanco en el resultado por ejemplo en caso de que exista un país que

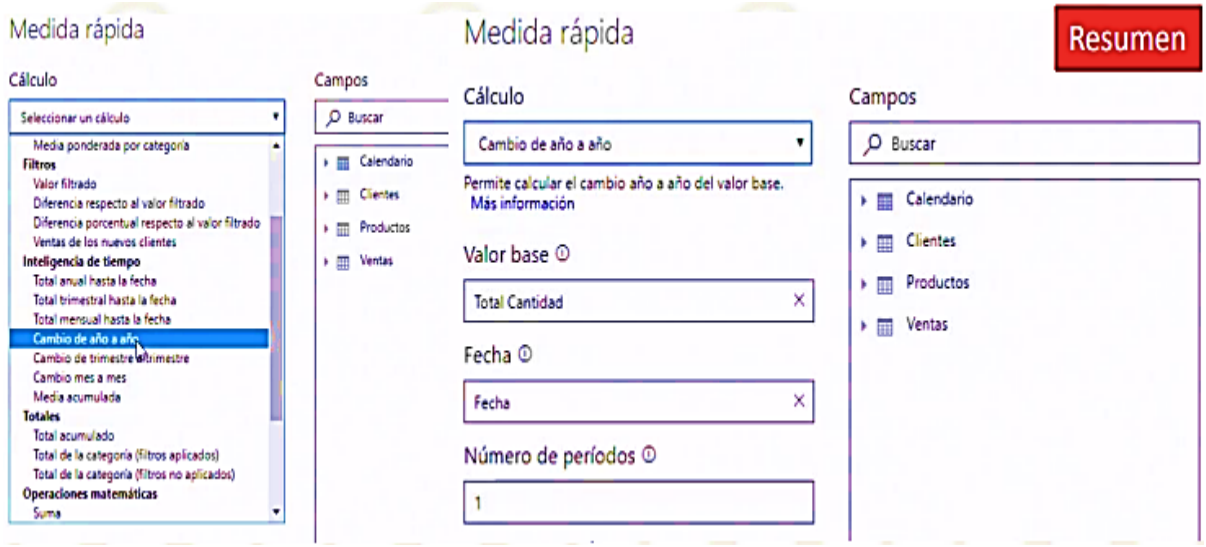
no tenga ventas o que se trate el dato como 0 y lo que vamos a comparar es las ventas de ese país igual a 0 contra las ventas en EEUU lo dejamos en generar espacio.

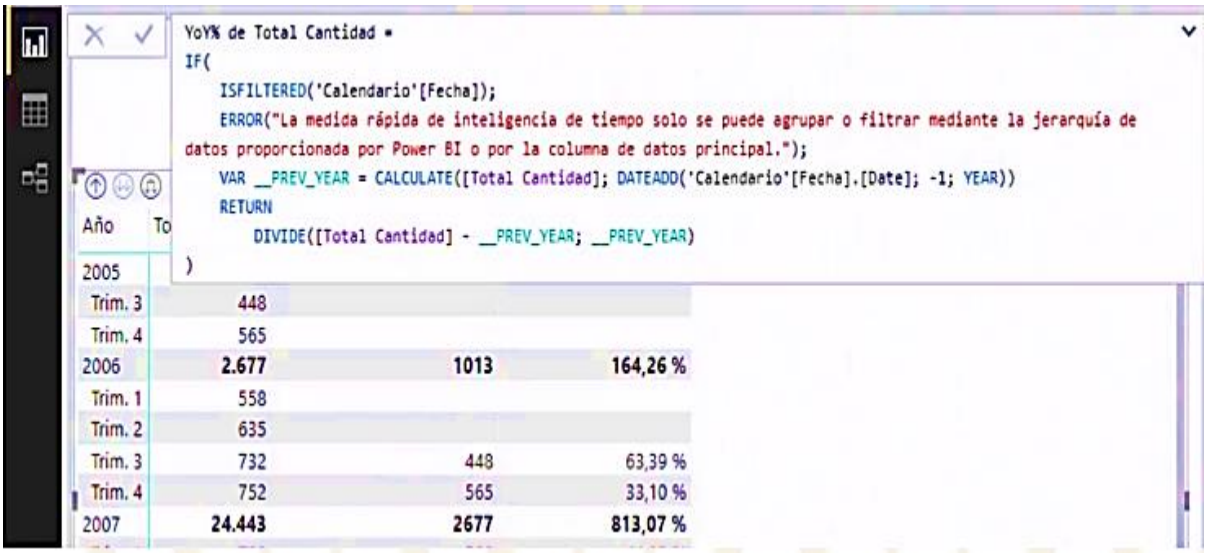


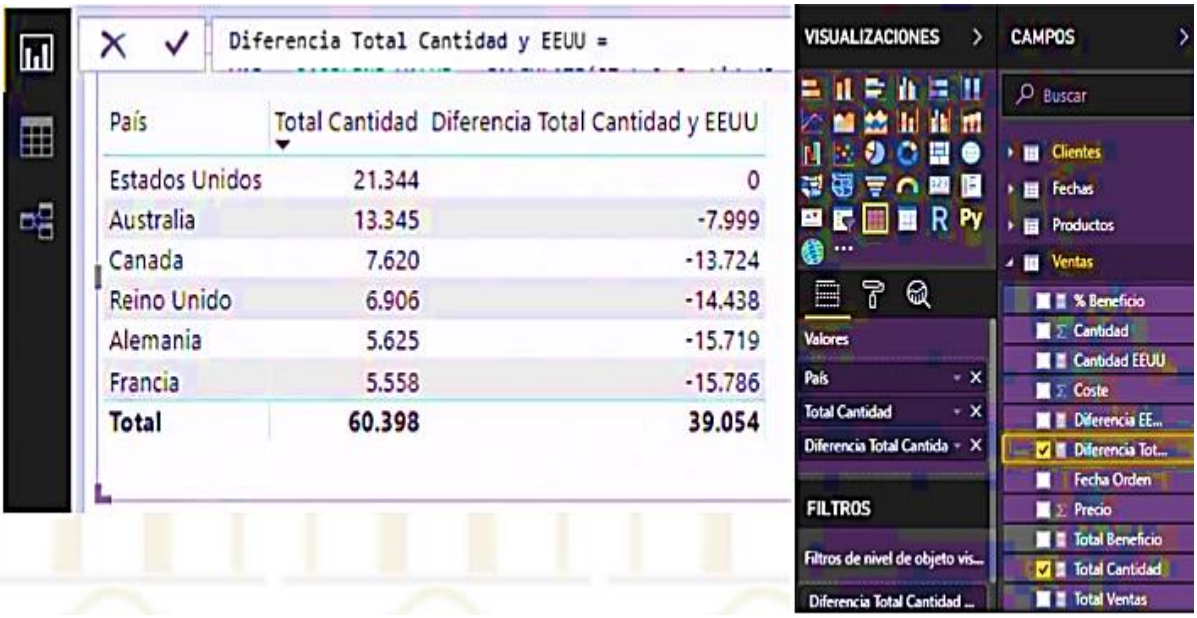
Para el filtro ya PBI ha escogido país que es la columna que estamos utilizando la tablas la columna y sólo nos queda seleccionar el valor en este caso seleccionar EEUU que es el país que queremos utilizar como referencia.

Ya todo configurado aceptamos aceptar y surge la magia, Power BI ha crea por nosotros el código DAX necesario para calcular la diferencia entre EEUU y el resto de los países agregando la medida a la tabla y además al modelo, así como nos muestra el código generado para crear la medida.




Medida rápida
Cálculo
Seleccionar un cálculo
Media ponderada por categoría
Filtros
Valor filtrado
Diferencia respecto al valor filtrado
Diferencia porcentual respecto al valor filtrado
Ventas de los nuevos clientes
Inteligencia de tiempo
Total anual hasta la fecha
Total trimestral hasta la fecha
Total mensual hasta la fecha
Cambio de año a año
Cambio de trimestre a trimestre
Cambio mes a mes
Media acumulada
Totales
Total acumulado
Total de la categoría (filtros aplicados)
Total de la categoría (filtros no aplicados)
Operaciones matemáticas
Suma
Campos
Buscar
Calendario
Clientes
Productos
Ventas
Medida rápida
Cálculo
Cambio de año a año
Permite calcular el cambio año a año del valor base.
Más información
Valor base
Total Cantidad
Fecha
Fecha
Número de períodos
1
Campos
Buscar
Calendario
Clientes
Productos
Ventas
Resumen


YoY% de Total Cantidad =
IF(
ISFILTERED('Calendario'[Fecha]);
ERROR("La medida rápida de inteligencia de tiempo solo se puede agrupar o filtrar mediante la jerarquía de datos proporcionada por Power BI o por la columna de datos principal.");
VAR __PREV_YEAR = CALCULATE([Total Cantidad]; DATEADD('Calendario'[Fecha].[Date]; -1; YEAR))
RETURN
DIVIDE([Total Cantidad] - __PREV_YEAR; __PREV_YEAR)
)
Año
2005
Trim. 3 448
Trim. 4 565
2006 2.677 1013 164,26 %
Trim. 1 558
Trim. 2 635
Trim. 3 732 448 63,39 %
Trim. 4 752 565 33,10 %
2007 24.443 2677 813,07 %


Diferencia Total Cantidad y EEUU =
País
Total Cantidad
Diferencia Total Cantidad y EEUU
Estados Unidos 21.344 0
Australia 13.345 -7.999
Canada 7.620 -13.724
Reino Unido 6.906 -14.438
Alemania 5.625 -15.719
Francia 5.558 -15.786
Total 60.398 39.054
VISUALIZACIONES
CAMPOS
Buscar
Clientes
Fechas
Productos
Ventas
% Beneficio
Cantidad
Cantidad EEUU
Coste
Diferencia EE...
Diferencia Tot...
Fecha Orden
Precio
Total Beneficio
Total Cantidad
Total Ventas
Valores
País
Total Cantidad
Diferencia Total Cantida
FILTROS
Filtros de nivel de objeto vis...
Diferencia Total Cantidad ...

El código siempre va a tener las funciones y expresiones que operan de la forma más eficiente para el rendimiento del informe. El código además hace uso de las variables en DAX son enormemente útiles y eficientes no debemos ignorar este código escrito por PBI porque representa un excelente recurso de aprendizaje del lenguaje.

Si consideramos cambiar el nombre de la medida lo podemos hacer directamente en el código o en la propia etiqueta y cambiarle el nombre.

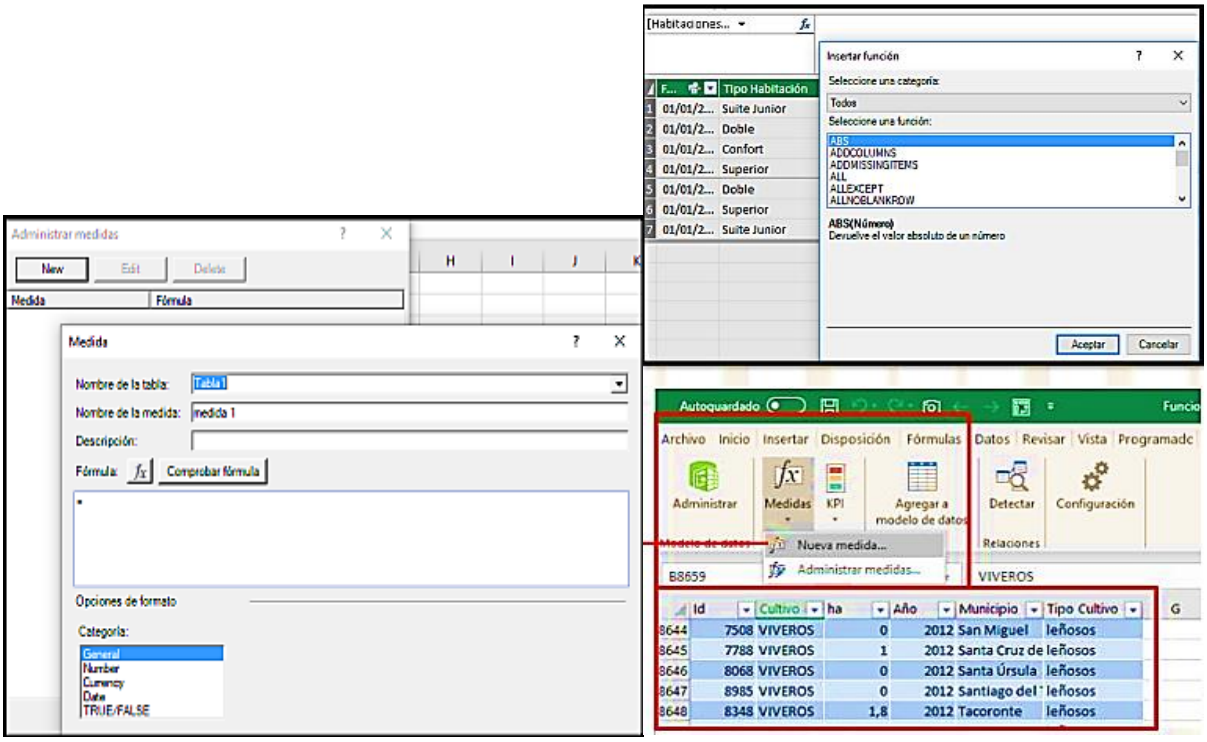
La ventaja es que el código DAX ha sido generado automáticamente, sin escribirlo nosotros.

Medidas rápidas en Power Pivot

Igualmente, en Power Pivot contamos con la opción de crear medidas rápidas y podremos acceder a ellas por diferentes vías entre ellas a través del icono Fx en la barra de fórmulas DAX de la ventana de Power Pivot que nos da acceso al cuadro de dialogo para introducir

las fórmulas en DAX y al cuadro de elección de las funciones disponibles agrupadas en categorías relacionadas en grupos afines según nuestra consideración.

En la pestaña Power Pivot contamos con la opción Administrar Medidas que nos permite, no solo crear sino además gestionar las medidas.



Consejos relacionados con la creación de medidas y columnas calculadas

No hay Columnas Desnudas en las medidas (sin función asignada)

Al hacer **referencia a una columna** en una fórmula de medida, siempre tiene que ser envuelta en una función.

Una referencia **desnuda** a una columna producirá un error en una medida.

Ejemplo:

[My New Measure] = Sales[Margin]… Al dar al botón comprobar formula nos da un error.

El valor de la columna 'Margin' en la tabla 'Sales' no puede ser determinado es necesario contar con una función, ... no puede estar desnuda. Todo lo siguiente sería válido:

=SUM(Sales[Margin]) …. =AVERAGE(Sales[Margin]) ….. =MIN(Sales[Margin]) …..=MAX(Sales[Margin])

Hacer referencia a las columnas desnudas es lo que las fórmulas de columnas calculadas hacen. Las medidas son agregaciones, y no aceptan referencias de columnas desnudas. Recuerde, las referencias de columnas desnudas son correcto en columnas calculadas. Esta regla sólo se aplica a las medidas.

Mejores Prácticas: Referenciar Columnas y Medidas de Manera Diferente

Cada vez que escribo una fórmula de medida, sigo estas dos técnicas:

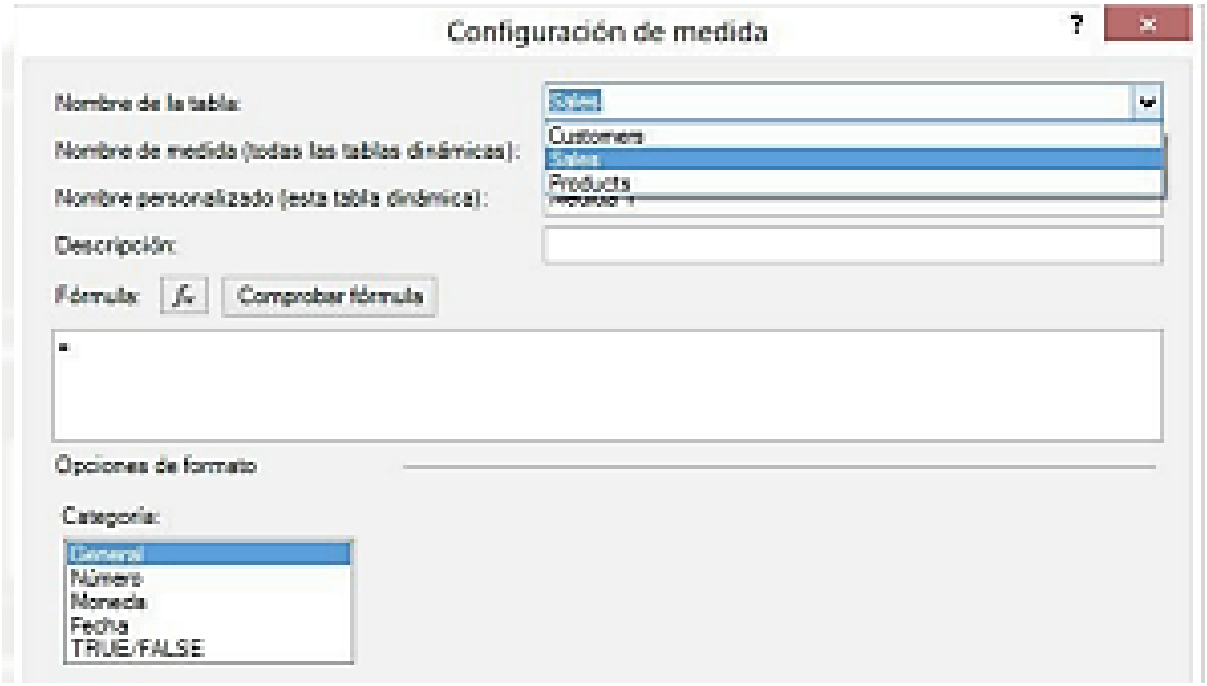
1. Para referenciar una columna, incluyo el nombre de la tabla:
   NombredeTabla [NombredeColumna]
2. Para referencia a una medida, omitimos el nombre de la tabla:
   [NombredeMedida]

Esto facilita que las fórmulas sean más legibles y permite diferenciar claramente si estamos haciendo referencia en una formula a una columna calculada o a una medida.

Además, hay muchas situaciones en las que la omisión del nombre de la tabla en una referencia de columna devolverá un error. Después de esta buena práctica evita ese problema también.

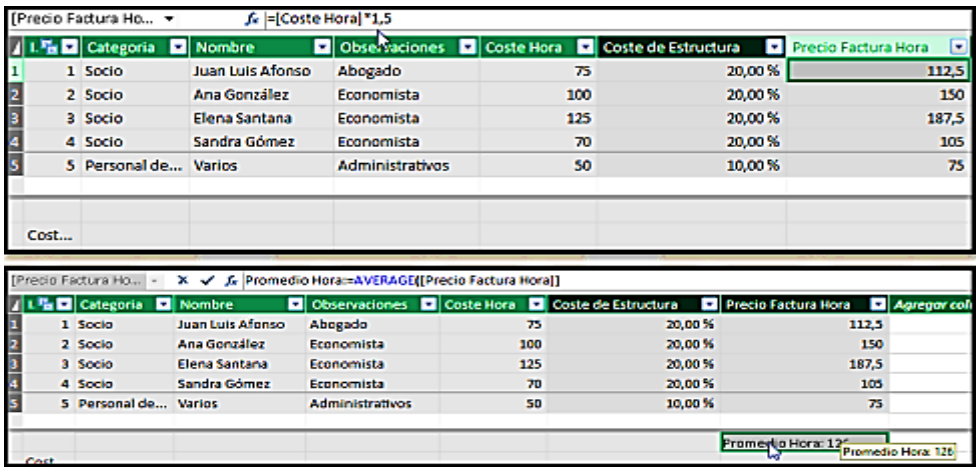
Mejores Prácticas: Asignar Medidas a las Tablas Correctas

El cuadro “nombre de tabla” en el editor de medida controla qué tabla se asignará la medida en la lista de campos. La medida será vinculada con la tabla de ventas en la lista de campos. **Regla Simple**: Asigno mis medidas a las tablas que contienen las columnas numéricas utilizadas en la fórmula. Esto no es más que una buena higiene para que nuestro modelo sea más fácil de entender más tarde.



Sintaxis para columna calculada (=) y campo (medida) calculado (:=)

El operador de asignación de funciones es distinto para columnas calculadas que para campos calculados. Para columna calculada utiliza el signo igual (=), mientras que para las medidas se utiliza dos puntos e igual (:=) después del nombre y antes de escribir la fórmula.



Referenciando a una tabla o una columna

En caso de tener que referenciar en una formula a una tabla y/o columna de nuestro modelo de datos el procedimiento a seguir es el siguiente:

Para escribir un nombre de una tabla. **Empiece a escribir el nombre de la tabla**. La función Autocompletar fórmula proporciona una lista desplegable que contiene nombres válidos que

comienzan con esas letras.

Para escribir el nombre de una columna. **Escriba un paréntesis y elija la columna en la lista de columnas de la tabla actual**. Para una columna de otra tabla, empiece a escribir las primeras letras del nombre de la tabla y, a continuación, elija la columna en la lista desplegable Autocompletar.

Elección entre columnas y medidas calculadas

Visto la diferencia entre las columnas calculadas y las medidas, es útil analizar cuándo usar una sobre la otra. A veces, cualquiera es una opción, pero en la mayoría de las situaciones, los requisitos de cálculo determinan la elección.

Como desarrollador, debe definir una columna calculada siempre que desee hacer lo siguiente:

- Coloque los resultados calculados en una cortadora o vea los resultados en las filas o columnas en una matriz o en una mesa pivotante, o utilice la condición de filtro de la columna calculada en una consulta DAX.
- Definitivamente una expresión que está estrictamente ligada a la corriente de entrada.

  Por ejemplo, Precio * Cantidad

  No se puede trabajar en una edad o en la suma de estas dos columnas.
- Categorizar texto o números. Por ejemplo, una gama de valores para una medida, una gama de edades de customers, como 0–18, 18–25, y así sucesivamente. Estas categorías se utilizan a menudo como filtros o para cortar valores.

Sin embargo, es obligatorio definir una medida siempre que se desee mostrar valores de cálculo que reflejen las selecciones del usuario, y los valores deben presentarse como agregados en un informe, por ejemplo:

- Calcular el porcentaje de profit de selección deportivo
- Para calcular las proporciones de un producto en comparación con todos los productos, mantenga el filtro tanto por año como por región.

Podemos expresar muchos cálculos tanto con columnas calculadas como con medidas, aunque necesitamos usar diferentes expresiones DAX para cada una. Por ejemplo, se puede definir el Margen bruto como columna calculada:

Ventas [GrossMargin] = Ventas [SalesAmount] - Ventas [TotalProductCost]

Sin embargo, también se puede definir como una medida:

GrossMargin: = SUM (Ventas [SalesAmount]) - SUM (Ventas [TotalProductCost])

Le sugerimos que utilice una medida en este caso porque, al ser evaluada en el momento de la consulta, no consume memoria ni espacio en disco. Como regla general, siempre que pueda expresar un cálculo en ambos sentidos, las medidas son la forma preferida de hacerlo. Debe limitar el uso de columnas calculadas a los pocos casos en que sean estrictamente necesarias. Los usuarios con experiencia en Excel suelen preferir las columnas calculadas a las medidas porque las columnas calculadas se parecen mucho a la forma de realizar cálculos en Excel. Sin embargo, la mejor forma de calcular un valor en DAX es a través de una medida.

Manejo de errores en expresiones Dax

Ahora que ha visto algunos de los conceptos básicos de la sintaxis, es hora de aprender a manejar con elegancia los cálculos no válidos. Una expresión DAX puede contener cálculos no válidos porque los datos a los que hace referencia no son válidos para la fórmula. Por ejemplo, la fórmula puede contener una división por cero o hacer referencia a un valor de columna que no es un número mientras se utiliza en una operación aritmética como la multiplicación. Es bueno aprender cómo se manejan estos errores de forma predeterminada y cómo interceptar estas condiciones para un manejo especial.

Sin embargo, antes de discutir cómo manejar los errores, describimos los diferentes tipos de errores que pueden aparecer durante la evaluación de una fórmula DAX. Son

- Conversión errores
- Los errores de operación aritmética.
- Empty o missing values

Errores de conversión

- El primer tipo de error es el error de conversión. DAX convierte automáticamente valores entre cadenas y números siempre que el operador lo requiera.

Todos estos ejemplos son expresiones DAX válidas:

"10" + 32 = 42

"10" y 32 = "1032"

10 y 32 = "1032"

FECHA (2010,3,25) = 25/03/2010 FECHA (2010,3,25) + 14 = 8/4/2010 FECHA (2010,3,25) & 14 = "25/03/201014"

Estas fórmulas siempre son correctas porque operan con valores constantes. Sin embargo, ¿qué pasa con la siguiente fórmula si Código del IVA es una cadena?

Ventas [VatCode] + 100

Debido a que el primer operando de esta suma es una columna que es de Texto tipo de datos, usted como desarrollador debe estar seguro de que DAX puede convertir todos los valores de esa columna en números. Si DAX no puede convertir parte del contenido para satisfacer las necesidades del operador, se producirá un error de conversión. A continuación, se muestran algunas situaciones típicas:

"1 + 1" + 0 = No se puede convertir el valor '1 + 1' del tipo Texto al tipo Número

DATEVALUE ("25/14/2010") = No coincide el tipo

Si desea evitar estos errores, es importante agregar lógica de detección de errores en las expresiones DAX para interceptar las condiciones de error y devolver un resultado que

tenga sentido. Se puede obtener el mismo resultado interceptando el error después de que haya ocurrido o verificando los operandos para la situación de error de antemano. No obstante, es mejor comprobar la situación de error de forma proactiva que dejar que suceda y luego detectarlo.

Errores de operaciones aritméticas

La segunda categoría de errores son las operaciones aritméticas, como la división por cero o la raíz cuadrada de un número negativo. Estos no son errores relacionados con la conversión:

DAX los genera cada vez que intentamos llamar a una función o usar un operador con valores no válidos.

La división por cero requiere un manejo especial porque su comportamiento no es intuitivo (excepto, quizás, para los matemáticos). Cuando uno divide un número por cero, DAX devuelve el valor especial infinito. En los casos especiales de 0 dividido por 0 o infinito dividido por infinito, DAX devuelve el especial Yaya (no es un número) valor.

Es importante tener en cuenta que infinito y Yaya no son errores sino valores especiales en DAX. De hecho, si uno divide un número por infinito, la expresión no genera ningún error. En

cambio, devuelve 0: 9954 / (7/0) = 0

Aparte de esta situación especial, DAX puede devolver errores aritméticos al llamar a una función con un parámetro incorrecto, como la raíz cuadrada de un número negativo:

SQRT (-1) = Un argumento de la función 'SQRT' tiene el tipo de datos incorrecto o el resultado es demasiado grande o demasiado pequeño.

Si DAX detecta errores como este, bloquea cualquier cálculo adicional de la expresión y genera un error. Uno puede usar el ISERROR función para comprobar si una expresión conduce a un error.

Tenga en cuenta que valores especiales como Yaya se muestran en la interfaz de usuario de varias herramientas, como Power BI, como valores regulares. Sin embargo, pueden ser tratados como errores cuando se muestran en otras herramientas cliente, como

una tabla dinámica de Excel. Finalmente, estos valores especiales son detectados como errores por las funciones de detección de errores.

Valores vacíos o faltantes

La tercera categoría que examinamos no es una condición de error específica, sino la presencia de valores vacíos. Los valores vacíos pueden generar resultados inesperados o errores de cálculo cuando se combinan con otros elementos en un cálculo.

DAX maneja los valores perdidos, los valores en blanco o las celdas vacías de la misma manera, usando el valor BLANCO. BLANCO no es un valor real, sino una forma especial de identificar estas condiciones. Podemos obtener el valor BLANCO en una expresión DAX llamando al BLANCO función, que es diferente de una cadena vacía. Por ejemplo, la siguiente expresión siempre devuelve un valor en blanco, que se puede mostrar como una cadena vacía o cómo "(en blanco)" en diferentes herramientas de cliente: = EN BLANCO ()

Por sí sola, esta expresión es inútil, pero la BLANCO la función en sí se vuelve útil cada vez que es necesario devolver un valor vacío. Por ejemplo, es posible que desee mostrar un resultado vacío en lugar de 0. La siguiente expresión calcula el descuento total para una transacción de venta, dejando el valor en blanco si el descuento es 0:

= SI (Ventas [DiscountPerc] = 0, - - Compruebe si hay un descuento EN BLANCO (), - - Devuelva un espacio en blanco si no hay descuento regalo

Ventas [DiscountPerc] * Ventas [monto] - - Calcule el descuento de lo contrario)

BLANCO, por sí mismo, no es un error; es solo un valor vacío. Por tanto, una expresión que contenga un BLANCO puede devolver un valor o un espacio en blanco, según el cálculo requerido. Por ejemplo, la siguiente expresión devuelve BLANCO cuando sea Cantidad de ventas] es BLANCO: = 10 * Ventas [Cantidad]

En otras palabras, el resultado de un producto aritmético es BLANCO siempre que uno o ambos términos sean BLANCO.

Esto crea un desafío cuando es necesario verificar un valor en blanco. Debido a las conversiones implícitas, es imposible distinguir si una expresión es 0 (o cadena vacía) o

BLANCO utilizando un operador igual. De hecho, las siguientes condiciones lógicas son siempre verdaderas:

EN BLANCO () = 0 - - Siempre devuelve VERDADERO

EN BLANCO () = "" - - Siempre devuelve VERDADERO

Por tanto, si las columnas Ventas [DiscountPerc] o Empleado de ventas] están en blanco, las siguientes condiciones vuelven CIERTO incluso si la prueba es contra 0 y cadena vacía,

respectivamente:

Sales [DiscountPerc] = 0 - Devuelve TRUE si DiscountPerc está EN BLANCO o 0

Sales [Clerk] = "" - - Devuelve TRUE si Clerk está EN BLANCO o ""

La propagación de BLANCO en una expresión DAX ocurre en varias otras operaciones aritméticas y lógicas, como se muestra en los siguientes ejemplos:

EN BLANCO () + EN BLANCO () = EN BLANCO () 10 * EN BLANCO () = EN BLANCO ()

EN BLANCO () / 3 = EN BLANCO () EN BLANCO () / EN BLANCO () = EN BLANCO ()

Sin embargo, la propagación de BLANCO en el resultado de una expresión no ocurre para todas las fórmulas.

Algunos cálculos no se propagan BLANCO. En cambio, devuelven un valor que depende de los otros términos de la fórmula. Ejemplos de estos son suma, resta, división por BLANCO, y una operación lógica que incluye un BLANCO. Las siguientes expresiones muestran algunas de estas condiciones junto con sus resultados:

BLANCO ()

18 + EN BLANCO () = 18

4 / EN BLANCO () = Infinito 0 / EN

BLANCO () = NaN

EN BLANCO () || EN BLANCO () = FALSO EN

BLANCO () && EN BLANCO () = FALSO (EN

BLANCO () = EN BLANCO ()) = VERDADERO (EN

BLANCO () = VERDADERO) = FALSO (EN

BLANCO () = FALSO) = VERDADERO (EN

BLANCO () = 0) = VERDADERO

(EN BLANCO () = "") = VERDADERO

ESBLANCO (EN BLANCO ()) = VERDADERO

FALSO || EN BLANCO () = FALSO FALSO &&

EN BLANCO () = FALSO VERDADERO || EN

BLANCO () = VERDADERO VERDADERO &&

EN BLANCO () = FALSO

Comprender el comportamiento de valores vacíos o faltantes en una expresión DAX y usar BLANCO devolver una celda vacía en un cálculo son habilidades importantes para controlar los resultados de una expresión DAX. Uno puede usar a menudo BLANCO como resultado al detectar valores incorrectos u otros errores, como demostramos en la siguiente sección.

Interceptar errores

Ahora que hemos detallado los diversos tipos de errores que pueden ocurrir, aún necesitamos mostrarle las técnicas para interceptar errores y corregirlos o, al menos, producir un mensaje de error que contenga información significativa. La presencia de errores en una expresión de DAX depende con frecuencia del valor de las columnas utilizadas en la propia expresión. Por lo tanto, es posible que desee controlar la presencia de estas condiciones de error y devolver un mensaje de error. La técnica estándar es verificar si una expresión devuelve un error y, de ser así, reemplazar el error con un mensaje específico o un valor predeterminado. Hay algunas funciones de DAX para esta tarea.

El primero de ellos es el SIERROR función, que es similar a la SI función, pero en lugar de evaluar una condición booleana, comprueba si una expresión devuelve un error. Dos usos típicos del SIERROR función son las siguientes:

= SIERROR (Ventas [Cantidad] * Ventas [Precio], EN BLANCO ())

= SIERROR (SQRT (Prueba [Omega]), EN BLANCO ())

En la primera expresión, si cualquiera Cantidad de ventas] o Precio de venta] es una cadena que no se puede convertir en un número, la expresión devuelta es un valor vacío. De lo contrario, el producto de Cantidad y Precio es regresado.

En la segunda expresión, el resultado es una celda vacía cada vez que Prueba [Omega] la columna contiene un número negativo.

Utilizando SIERROR de esta manera corresponde a un patrón más general que requiere el uso ISERROR y SI:

= SI (IS.ERROR (Ventas [Cantidad] * Ventas [Precio]), BLANCO (), Ventas [Cantidad] * Ventas [Precio])

= SI (ISERROR (SQRT (Prueba [Omega])), EN BLANCO (), SQRT (Prueba [Omega]))

En estos casos, SIERROR es una mejor opción. Uno puede usar SIERROR siempre que el resultado sea la misma expresión probada para detectar un error; no es necesario duplicar la expresión en dos lugares, y el código es más seguro y legible.

Sin embargo, un desarrollador debe usar SI cuando quieren devolver el resultado de una expresión diferente.

Además, se puede evitar la aparición del error probando los parámetros antes de usarlos. Por ejemplo, se puede detectar si el argumento a favor SQRT es positivo, regresando BLANCO para valores negativos:

= SI (Prueba [Omega]> = 0, SQRT (Prueba [Omega]), EN BLANCO ())

Considerando que el tercer argumento de un SI declaración por defecto a BLANCO, también se puede escribir la misma expresión de forma más concisa:

= SI (Prueba [Omega]> = 0, SQRT (Prueba [Omega]))

Un escenario frecuente es probar con valores vacíos.

ESTÁ EN BLANCO detecta valores vacíos, volviendo CIERTO si su argumento es BLANCO. Esta capacidad es importante especialmente cuando un valor no disponible no implica que sea 0. El siguiente ejemplo calcula el costo de envío de una transacción de venta, utilizando un costo de envío predeterminado para el producto si la transacción en sí no especifica un peso:

= SI (ISBLANK (Ventas [Peso]), - - Si falta el peso Sales [DefaultShippingCost], - - luego devuelva el costo predeterminado Ventas [Peso] * Ventas [Precio de envío] - - de lo contrario, multiplique el peso por envío precio).

Si simplemente multiplicamos el peso del producto por el precio de envío, obtenemos un costo vacío para todas las transacciones de ventas sin datos de peso debido a la propagación de BLANCO en multiplicaciones.

Cuando se usan variables, los errores deben verificarse en el momento de la definición de la variable en lugar de donde los usamos. De hecho, la primera fórmula en el siguiente código devuelve cero, la segunda fórmula siempre arroja un error y la última produce resultados diferentes según la versión del producto que usa DAX (la última versión también arroja un error):

SIERROR (SQRT (-1), 0) - - Esto devuelve 0

VAR WrongValue = SQRT (-1) - - Aquí ocurre un error, por lo que el resultado es

REGRESO - - siempre un error

IFERROR (WrongValue, 0) - - Esta línea nunca se ejecuta

SIERROR (- - Resultados diferentes según versiones VAR WrongValue = SQRT (-1)

- - IFERROR arroja un error en 2017 versiones

REGRESO

- - IFERROR devuelve 0 en las versiones hasta 2016 Valor incorrecto, 0)

El error ocurre cuando Valor incorrecto se evalúa.

Por lo tanto, el motor nunca ejecutará SIERROR en el segundo ejemplo, mientras que el resultado del tercer ejemplo depende de las versiones del producto. Si necesita verificar si hay errores, tome algunas precauciones adicionales al usar variables.

Generando errores

A veces, un error es solo un error y la fórmula no debe devolver un valor predeterminado en caso de error. De hecho, devolver un valor predeterminado terminaría produciendo un resultado real que sería incorrecto. Por ejemplo, una tabla de configuración

que contiene datos inconsistentes debería producir un informe no válido en lugar de números que no son confiables y, sin embargo, podría considerarse correcto.

Además, en lugar de un error genérico, es posible que desee generar un mensaje de error que sea más significativo para los usuarios. Tal mensaje ayudaría a los usuarios a encontrar dónde está el problema.

Considere un escenario que requiere el cálculo de la raíz cuadrada de la temperatura absoluta medida en Kelvin, para ajustar aproximadamente la velocidad del sonido en un cálculo científico complejo. Obviamente, no esperamos que la temperatura sea un número negativo. Si eso sucede debido a un problema en la medición, debemos generar un error y detener el cálculo.

En ese caso, este código es peligroso porque oculta el problema:

= SIERROR (SQRT (Prueba [Temperatura]), 0)

En cambio, para proteger los cálculos, se debe escribir la fórmula de esta manera:

= SI (Prueba [Temperatura]> = 0, SQRT (Prueba [Temperatura]), ERROR ("La temperatura no puede ser un número negativo. Cálculo cancelado."))

Tipos de datos DAX

En una expresión DAX, el tipo resultante, se basa en el tipo de término utilizado en la expresión. Debe tener en cuenta esto en caso de que el tipo devuelto por una expresión DAX no sea el tipo esperado; entonces tendría que investigar el tipo de datos de los términos usados en la expresión misma.

Entero

DAX solo tiene uno Entero tipo de datos que puede almacenar un valor de 64 bits. Todos los cálculos internos entre valores enteros en DAX también utilizan un valor de 64 bits.

Decimal

A Decimal El número siempre se almacena como un valor de punto flotante de doble precisión. No confunda este tipo de datos DAX con el decimal y numérico tipo de datos de Transact-SQL. El tipo de datos correspondiente de un número decimal DAX en SQL es Flotador.

Divisa

los Divisa tipo de datos, también conocido como Número decimal fijo en Power BI, almacena un número decimal fijo. Puede representar cuatro puntos decimales y se almacena internamente como un valor entero de 64 bits dividido por 10.000. Sumar o restar Divisa los tipos de datos siempre ignoran los decimales más allá del cuarto punto decimal, mientras que la multiplicación y la división producen un valor de punto flotante, aumentando así la precisión del resultado. En general, si necesitamos más precisión que los cuatro dígitos proporcionados, debemos usar un Decimal tipo de datos.

El formato predeterminado del Divisa el tipo de datos incluye el símbolo de moneda. También podemos aplicar el formato de moneda a Entero y números decimales, y podemos usar un formato sin el símbolo de moneda para un Divisa tipo de datos.

Fecha y hora

DAX almacena las fechas como Fecha y hora tipo de datos. Este formato utiliza internamente un número de coma flotante, en el que el número entero corresponde al número de días desde el 30 de diciembre de 1899 y la parte decimal identifica la fracción del día. Las horas, los minutos y los segundos son convertido a fracciones decimales de un día. Por lo tanto, la siguiente expresión devuelve la fecha actual más un día (exactamente 24 horas):

= HOY () + 1

El resultado es la fecha de mañana en el momento de la evaluación. Si necesita tomar solo la fecha parte de un Fecha y hora, recuerda siempre usar TRUNC para deshacerse de la parte decimal.

Power BI ofrece dos tipos de datos adicionales: Fecha y Tiempo. Internamente, son una simple variación de Fecha y hora. En efecto, Fecha y Tiempo almacenar solo el entero o la parte decimal del Fecha y hora, respectivamente.

Booleano

Los Booleano. El tipo de datos se utiliza para expresar condiciones lógicas. Por ejemplo, una columna calculada definida por la siguiente expresión es de Booleano escribe:

= Ventas [Precio unitario]> Ventas [Costo unitario]

Veras, Booleanos tipos de datos como números donde CIERTO es igual a 1 y FALSO es igual a 0. Esta notación a veces resulta útil para fines de clasificación porque CIERTO > FALSO.

Cadena

Cada cadena en DAX se almacena como un Unicode cadena, donde cada carácter se almacena en 16 bits. De forma predeterminada, la comparación entre cadenas no distingue entre mayúsculas y minúsculas, por lo que las dos cadenas Power BI y POWER BI se consideran iguales.

Variante

Este tipo de datos se utiliza para expresiones que pueden devolver diferentes tipos de datos, según las condiciones. Por ejemplo, la siguiente declaración puede devolver un número entero o una cadena, por lo que devuelve un tipo de variante:

SI ([medida]> 0, 1, "N / A")

Binario

El tipo de datos binario, se utiliza en el modelo de datos para almacenar imágenes u otros tipos de información no estructurada. No está disponible en DAX. Fue utilizado principalmente por Power View, pero es posible que no esté disponible en otras herramientas como Power BI.

Operadores de Dax

Aritméticos:

Según (Escuela De Postgrado Y Educación Continua, Facultad De Ciencias Físicas y Matemáticas, Universidad de Chile, s.f.), las fórmulas de DAX pueden usar operadores de DAX para crear expresiones que puedan realizar cálculos aritméticos, comparaciones o probar condiciones. Primero son los operadores aritméticos que son mostrados en Figura 27.

Figura 27

Operadores de Dax

Operador	Descripción
+	Suma
-	Resta
*	Multiplicación
/	División
^	Exponenciación

Fuente: (Escuela De Postgrado Y Educación Continua, Facultad De Ciencias Físicas y Matemáticas, Universidad de Chile, s.f.)

Una función relevante que vale la pena destacar es la función DIVIDE la cual reemplaza al operador de división (/), la cual puede devolver un valor predeterminado o BLANK cuando ocurre una división por cero.

Comparación:

En segundo lugar, están los operadores de comparación se utilizan para comparar dos valores, siendo el resultado de dicha comparación TRUE si es verdadero o FALSE si es falsa. En la Figura 28 se enumeran los operadores de comparación.

Figura 28

Operadores de comparación

Operador	Descripción
=	Igual a
==	Estrictamente igual a
>	Mayor que
<	Menor que
>=	Mayor o igual que
<=	Menor o igual que
<>	No igual a

Fuente: (Escuela De Postgrado Y Educación Continua, Facultad De Ciencias Físicas y Matemáticas, Universidad de Chile, s.f.)

Es importante destacar que todos los operadores de comparación, excepto == consideran BLANK como igual al número cero, una cadena vacía (“”), la fecha 30 de diciembre o FALSE. Esto significa que la expresión [Sales Amount] = 0 será TRUE cuando el valor de [Sales Amount] sea cero o BLANK. En cambio, [Sales Amount] == 0] es TRUE solo cuando el valor de [Sales Amount] es cero.

Lógicos:

En la Figura 29 los operadores lógicos que permiten combinar expresiones para producir un

único resultado.

Figura 29

Operadores lógicos

Operador	Descripción
&& (doble ampersand)	Crea una condición AND entre dos expresiones
|| (doble pipe)	Crea una condición OR entre dos expresiones
IN	Crea una condición OR entre cada fila a ser compara con la tabla

Fuente: (Escuela De Postgrado Y Educación Continua, Facultad De Ciencias Físicas y Matemáticas, Universidad de Chile, s.f.)

CAPÍTULO TRES

FUNCIONES DAX

Introducción

Las Expresiones de Análisis de Datos (DAX) son un lenguaje de expresiones de fórmulas que se usa en Analysis Services, Power BI y Power Pivot en Excel. Las fórmulas DAX abarcan funciones, operadores y valores para realizar cálculos avanzados y consultas en los datos de las tablas y columnas relacionadas de los modelos de datos tabulares.

Las fórmulas DAX se usan en medidas, columnas calculadas, tablas calculadas y seguridad de nivel de fila. Las medidas son fórmulas de cálculo dinámico en las que los resultados cambian en función del contexto. Las medidas se usan en informes en los que se pueden combinar y filtrar datos del modelo mediante varios atributos, como un informe de Power BI o una tabla dinámica o un gráfico dinámico de Excel.

DAX es una recopilación de funciones, operadores y constantes que se pueden usar en una fórmula o expresión para calcular y devolver uno o varios valores. DAX ayuda a crear información de datos nueva que ya está en un modelo. Las fórmulas de DAX proporcionan esta capacidad y muchas otras funciones importantes.

Funciones Dax

Las funciones son fórmulas predefinidas que realizan cálculos por medio de valores específicos, denominados argumentos, en un orden o estructura determinados. Los argumentos pueden ser otras funciones, otra fórmula, una expresión, referencias de columna, números, texto, valores lógicos como TRUE o FALSE, o constantes.

Las funciones DAX están divididas en las siguientes categorías (Cárdenas, 2021):

- Funciones Lógicas
- Funciones Matemáticas
- Funciones Estadísticas
- Funciones de Fecha y Hora
- Funciones de Texto
- Funciones de Filtrado

- Funciones de Inteligencia de Tiempo
- Funciones de Información
- Funciones Primarias y Secundarias
- Otras Funciones

Si estás familiarizado con las funciones en las fórmulas de Excel, muchas de las funciones DAX le parecerán semejantes; sin embargo, las funciones DAX son únicas de las maneras siguientes:

Una función de DAX siempre hace referencia a una columna completa o una tabla.

- Si solo desea usar valores concretos de una tabla o columna, puede agregar filtros a la fórmula.
- Si necesita personalizar los cálculos fila por fila, DAX ofrece funciones que le permiten usar el valor de la fila actual o un valor relacionado como un tipo de argumento para realizar cálculos en función del contexto. Aprenderá más acerca del contexto más adelante.
- DAX incluye muchas funciones que devuelven una tabla en lugar de un valor. La tabla no se muestra, pero se usa para proporcionar datos para otras funciones. Por ejemplo, puede recuperar una tabla y, a continuación, contar los valores distintos que contiene, o calcular sumas dinámicas en tablas o columnas filtradas.
- DAX incluye varias funciones de inteligencia de tiempo. Estas funciones permiten definir o seleccionar intervalos de fechas y realizar cálculos dinámicos con base en ellos. Por ejemplo, puede comparar sumas en períodos paralelos.
- Excel tiene una función popular, BUSCARV. Las funciones DAX no toman una celda o rango de celdas como una referencia como BUSCARV en Excel. Las funciones DAX toman una columna o una tabla como referencia. Tenga en cuenta que, en Power BI Desktop, está trabajando con un modelo de datos relacionales. Buscar valores en otra tabla es fácil y en la mayoría de los casos no es necesario crear ninguna fórmula.

Como se puede ver, las funciones de DAX pueden ayudarle a crear fórmulas eficaces.

A medida que desarrolle sus habilidades de DAX, podrá crear fórmulas con muchas funciones diferentes. Uno de los mejores lugares para obtener información detallada sobre cada una de las funciones de DAX es la referencia de funciones de DAX.

A continuación, encontrarás las funciones más utilizadas en la primera tabla de cada categoría, seguido por una tabla con todas las funciones de la categoría.

Categoría: Funciones Lógicas

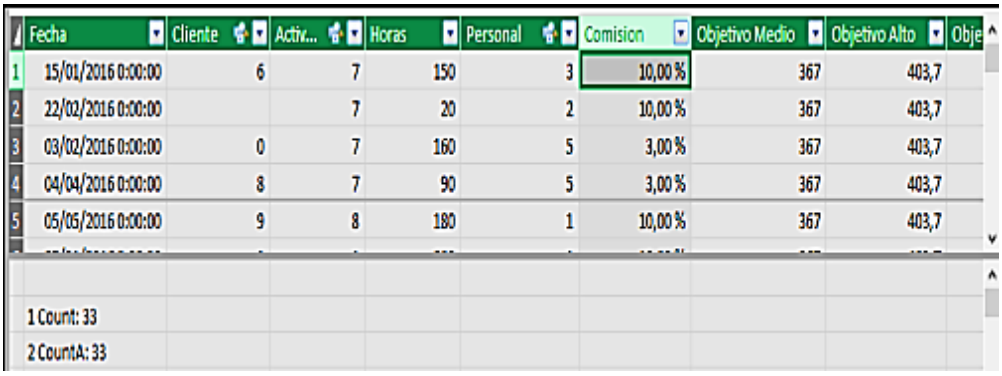
Funciones más utilizadas:

Función	Descripción	Sintaxis
IF	Revisa si se cumple con una condición. Si la condición se cumple, regresa un valor. Si la condición no se cumple, regresa otro valor.	**IF**(Condición, Resultado si es verdadera, Resultado si es falsa)
IFERROR	Evalúa una expresión. Si la expresión arroja un error, la función nos da un valor que hayamos definido. Si la expresión es correcta, regresará el valor de la expresión.	**IFERROR**(Expresión, Expresión si es error)
AND	Revisa si dos argumentos son verdaderos. Si ambos son verdaderos, regresa TRUE. Si uno o ambos son falsos, regresa FALSE.	**AND**(Expresión1, Expresión 2)
OR	Revisa si dos argumentos son verdaderos. Si uno o ambos son verdaderos, regresa TRUE. Si ambos son falsos, regresa FALSE.	**OR**(Expresión1, Expresión 2)

Ejemplo: Uso de IF con RELATED

En el siguiente caso vamos a crear una nueva columna calculada en la tabla D-Partes de nuestro modelo de datos, que en función de la categoría del empleado asigne un porcentaje de comisión, si es Socio el 10% y si es Personal de Apoyo el 3%. Por tanto, tendremos que establecer una formula en la columna y en función del código del empleado me consulte en la tabla Personal que categoría tiene y en base a ello aplique la comisión correspondiente.

De esta forma y en función de las condiciones establecidas la formula a insertar en la nueva columna calculada será la siguiente:



	Fecha	Cliente	Activ...	Horas	Personal	Comision	Objetivo Medio	Objetivo Alto	Obje
1	15/01/2016 0:00:00	6	7	150	3	10,00 %	367	403,7	
2	22/02/2016 0:00:00		7	20	2	10,00 %	367	403,7	
3	03/02/2016 0:00:00	0	7	160	5	3,00 %	367	403,7	
4	04/04/2016 0:00:00	8	7	90	5	3,00 %	367	403,7	
5	05/05/2016 0:00:00	9	8	180	1	10,00 %	367	403,7	

1 Count: 33
2 CountA: 33

```
=IF(RELATED(Tabla_A_personal[Categoria])="Socio";0,1;0,03)
```

Se podría leer la fórmula de la siguiente forma:

Relaciona la tabla Personal y su columna Categoría con el valor de la Columna Personal de la tabla Partes y si el valor de correspondencia es Socio, en ese caso le asignas la comisión del 10% y en caso contrario le asigna un 3%. En este caso se utiliza la propiedad RELATED para relacionar campos de otras tablas relacionadas.

Más funciones en la categoría:

Función	Descripción	Sintaxis
FALSE	Devuelve el valor lógico FALSE	**FALSE()**
NOT	Cambia FALSE por TRUE y TRUE por FALSE	**NOT**(expresión)
SWITCH	Devuelve resultados diferentes en función de una expresión	**SWITCH**(Expresión, Valor1, Resultados1, ..., [Else])
TRUE	Devuelve el valor lógico TRUE	**TRUE()**

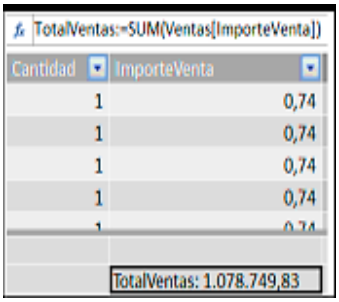
Categoría: Funciones Matemáticas

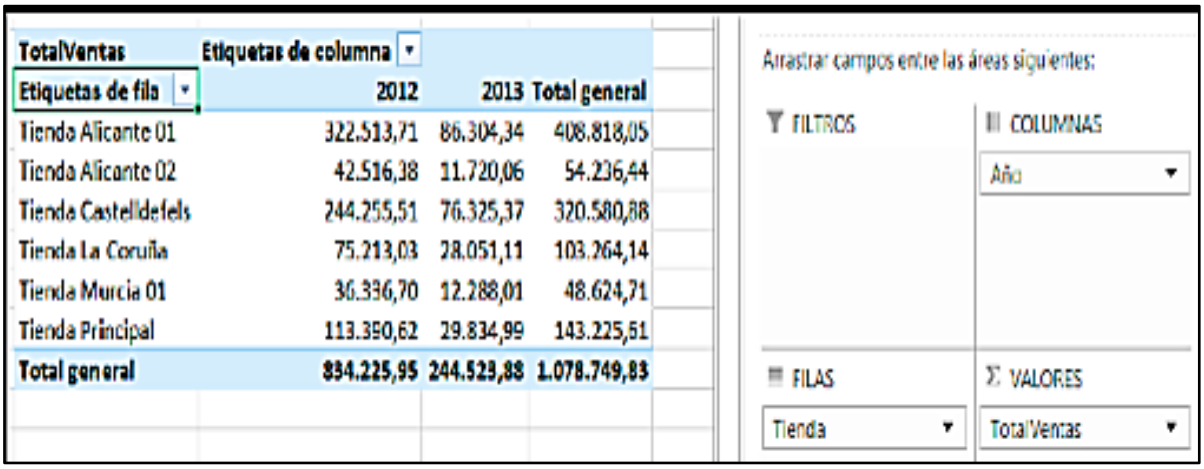
Funciones más utilizadas:

Función	Descripción	Sintaxis
SUM	Suma todos los números de una columna	**SUM**(Nombre de la columna)
ROUND	Redondea un número al número de dígitos especificado	**ROUND**(Número, Número de dígitos)
RAND	Devuelve un número aleatorio mayor o igual que 0 y menor que 1, distribuido uniformemente. Los números aleatorios cambian en los recálculos.	**RAND()**
RANDBETWEEN	Devuelve un número aleatorio entre los números que especifique	**RANDBETWEEN**(Abajo, Arriba)
DIVIDE	Función de división segura con posibilidad de controlar divisiones por cero	**DIVIDE**(Numerador, Denominador, [Resultado Alterno])

Ejemplo:

Queremos calcular el total de ventas de la empresa, crearemos una columna calculada utilizando la función SUM(): Esta función vista nos ofrece la suma de todos los valores de la columna “[ImporteVenta]” y le aplicamos el formato adecuado. Si creamos una tabla dinámica arrastramos “TotalVentas” al área de valores, nos muestra el resultado de sumar “[ImporteVenta]” para todas las filas de la tabla y ya formateado. Ahora lo desglosamos por Tienda y una columna por cada año:





Más funciones en la categoría:

Función	Descripción	Sintaxis
ABS	Devuelve el valor absoluto de un número	ABS(Número)
ACOS	Devuelve el arcoseno o el coseno inverso de un número. El arcoseno es el ángulo cuyo coseno es un número. El ángulo devuelto se proporciona en radianes en el rango de 0 (cero) a pi.	ACOS(Número)
ACOSH	Devuelve el coseno hiperbólico inverso de un número. El número debe ser mayor o igual que 1. El coseno hiperbólico inverso es el valor cuyo coseno hiperbólico es número, por lo que ACOSH(COSH(número)) es igual a un número.	ACOSH(Número)
ASIN	Devuelve el arcoseno o el seno inverso de un número. El arcoseno es el ángulo cuyo seno es un número. El ángulo devuelto se proporciona en radianes en el rango de -pi/2 pi/2.	ASIN(Número)

ASINH	Devuelve el seno hiperbólico inverso de un número. El seno hiperbólico inverso es el valor cuyo seno hiperbólico es número, por lo que ASINH(SINH(número)) es igual a un número.	**ASINH**(Número)
ATAN	Devuelve el arcotangente o la tangente inversa de un número. El arcotangente es el ángulo cuya tangente es un número. El ángulo devuelto se proporciona en radianes en el rango de -pi/2 a pi/2.	**ATAN**(Número)
ATANH	Devuelve la tangente hiperbólica inversa de un número. El número debe ser entre -1 y 1 (excluyendo -1 y 1). La tangente hiperbólica es un número, de modo que ATANH(TANH(número)) es igual a un número.	**ATANH**(Número)
CEILING	Redondea un número hacia arriba, a la unidad significativa o el entero más próximo.	**CEILING**(Número, Cifra significativa)

COMBIN	Devuelve el número de combinaciones de un número dado de elementos. Use COMBIN para determinar el número total posible de grupos de un número dado de elementos.	**COMBIN**(Número, Number_chosen)
COMBINA	Devuelve la cantidad de combinaciones (con repeticiones) de una cantidad determinada de elementos.	**COMBINA**(Número, Number_chosen)
COS	Devuelve el coseno del ángulo dado.	**COS**(Número)
COSH	Devuelve el coseno hiperbólico de un número.	**COSH**(Número)
CURRENCY	Devuelve el valor como un tipo de datos de moneda.	**CURRENCY**(Valor)
DEGREES	Convierte los radianes en grados.	**DEGREES**(Ángulo)
EVEN	Devuelve el número redondeado hacia arriba al entero par más cercano. Puede usar esta función para procesar los elementos que vienen en dos. Por ejemplo, un contenedor de empaquetado acepta filas de uno o dos artículos.	**EVEN**(Número)
EXP	Devuelve e elevado a la potencia de un número determinado.	**EXP**(Número)

FACT	Devuelve el factorial de un número, igual a 1*2*3*...* Número.	**FACT**(Número)
FLOOR	Redondea un número hacia abajo, hacia cero, al múltiplo significativo más cercano.	**FLOOR**(Número, Cifra significativa)
GCD	Devuelve el máximo común divisor de dos enteros, que es el entero mayor que divide tanto el número1 como el número2 sin un resto.	**GCD**(Número1, Número2)
INT	Redondea un número hasta el entero inferior más próximo.	**INT**(Número)
ISO.CEILING	Redondea un número hacia arriba, al entero o al múltiplo significativo más próximo.	**ISO.CEILING**(Número, [Cifra significativa])

LCM	Devuelve el mínimo común múltiplo de enteros, que es menor entero positivo que es múltiplo tanto del argumento de entero número1 como del número2. Use LCM para agregar fracciones con diferentes denominadores.	**LCM**(Número1, Número2)
LN	Devuelve el logaritmo natural de un número.	**LN**(Número)
LOG	Devuelve el logaritmo de un número en la base especificada.	**LOG**(Número, [Base])
LOG10	Devuelve el logaritmo en base 10 de un número.	**LOG10**(Número)

MOD	Devuelve el resto tras dividir un número por un divisor.	**MOD**(Número, Divisor)
MROUND	Devuelve un número redondeado al múltiplo deseado.	**MROUND**(Número, Multiple)
ODD	Devuelve un número redondeado al entero impar más cercano.	**ODD**(Número)
PI	Devuelve el valor Pi (3,14159265358979) con precisión de 15 dígitos.	**PI()**
POWER	Devuelve el resultado de elevar un número a una potencia.	**POWER**(Número, Power)
PRODUCTX	Devuelve el producto de los valores de una expresión en una tabla.	**PRODUCTX**(Tabla, Expresión)

QUOTIENT	Devuelve la parte entera de una división.	**QUOTIENT**(Numerador, Denominador)
RADIANS	Convierte los grados en radianes.	**RADIANS**(Número)
ROUNDDOWN	Redondea un número hacia abajo, hacia cero.	**ROUNDDOWN**(Número, NumberOfDigits)
ROUNDUP	Redondea un número hacia arriba, en dirección contraria a cero.	**ROUNDUP**(Número, NumberOfDigits)

SIGN	Devuelve el signo de un número: 1, si el número es positivo; cero, si el número es cero y -1, si el número es negativo.	**SIGN**(Número)
SQRT	Devuelve la raíz cuadrada de un número.	**SQRT**(Número)
SUMX	Devuelve la suma de una expresión evaluada para cada fila de una tabla.	**SUMX**(Tabla, Expresión)
TRUNC	Convierte un número decimal en uno entero al quitar la parte decimal o de fracción.	**TRUNC**(Número, [NumberOfDigits])

Categoría: Funciones Estadísticas

Funciones más utilizadas:

Función	Descripción	Sintaxis
AVERAGE	Devuelve el promedio (media aritmética) de todos los números de una columna	**AVERAGE**(ColumnName)
COUNT	Cuenta los números de una columna	**COUNT**(ColumnName)
COUNTA	Cuenta el número de valores de una columna	**COUNTA**(ColumnName)
DISTINCTCOUNT	Cuenta el número de valores distintos de una columna.	**DISTINCTCOUNT**(ColumnName)
MAX	Devuelve el valor numérico mayor de una columna o el valor mayor de dos expresiones escalares. Omite los valores lógicos y el texto.	**MAX**(ColumnNameOrScalar1, [Scalar2])
MIN	Devuelve el valor numérico menor de una columna o el valor menor de dos expresiones escalares. Omite los valores lógicos y el texto.	**MIN**(ColumnNameOrScalar1, [Scalar2])

Más funciones en la categoría:

Función	Descripción	Sintaxis
ADDCOLUMNS	Devuelve una tabla con nuevas columnas especificados por expresiones DAX.	**ADDCOLUMNS**(Tabla, Nombre1, Expresión1, ...)
APPROXIMATEDISTINCTCOUNT		**APPROXIMATEDISTINCTCOUNT**
AVERAGEA	Devuelve el promedio (media aritmética) de los valores de una columna. Procesa el texto y los valores no numéricos.	**AVERAGEA**(ColumnName)
AVERAGEX	Devuelve el promedio (media aritmética) de un conjunto de expresiones evaluadas en una tabla.	**AVERAGEX**(Tabla, Expresión)
BETA.DIST	Devuelve la distribución beta se usa normalmente para estudiar la variación en el porcentaje de algo a través de muestras, como la parte del día que la gente empleó en ver la televisión.	**BETA.DIST**(X, Alfa, Beta, Acumulativo, [A], [B])
BETA.INV	Devuelve la inversa de la función de densidad de probabilidad acumulativa (BETA.DIST). Si probability = BETA.DIST(x,...TRUE), entonces BETA.INV (probability,...) = x. La distribución beta puede usarse en la planificación de proyectos para modelar tiempos de finalización probables, dados una variabilidad y un tiempo de finalización esperados.	**BETA.INV**(Probabilidad, Alfa, Beta, [A], [B])
CHISQ.INV	Devuelve el inverso de la probabilidad de cola izquierda de la distribución chi cuadrado, que se usa habitualmente para	**CHISQ.INV**(Probabilidad, Grados_de_libertad)
	estudiar la variación en el porcentaje de algo en muestras, como la fracción del día que las personas pasan viendo la televisión.	
CHISQ.INV.RT	Devuelve el inverso de la probabilidad de cola derecha de la distribución chi cuadrado. Si probabilidad = CHISQ.DIST.RT(x,...) CHISQ.INV.RT (probability,...) = x. Use esta función para comparar los resultados observados con los esperados para decidir si su hipótesis original es válida.	**CHISQ.INV.RT** Probabilidad, Grados_de_libertad)

CONFIDENCE.NORM	Devuelve el intervalo de confianza para la media de probación, con una distribución normal.	**CONFIDENCE.NORM**(Alfa, Desv_estándar, Tamaño)
CONFIDENCE.T	Devuelve el intervalo de confianza para una media de probación, con una distribución t de Estudiante.	**CONFIDENCE.T**(Alfa, Desv_estándar, Tamaño)
COUNTAX	Cuenta el número de valores que son el resultado de evaluar una expresión para cada fila de una tabla.	**COUNTAX**(Tabla, Expresión)
COUNTBLANK	Cuenta el número de espacios en blanco de una columna.	**COUNTBLANK**(ColumnName)
COUNTROWS	Cuenta el número de filas de una tabla.	**COUNTROWS**(Tabla)
COUNTX	Cuenta el número de valores que son el resultado de evaluar una expresión para cada fila de una tabla.	**COUNTX**(Tabla, Expresión)

CROSSJOIN	Devuelve una tabla que es una combinación cruzada de las tablas especificadas.	**CROSSJOIN**(Tabla1, ...)

DATATABLE	Devuelve una tabla con datos definidos en línea.	**DATATABLE**(nombre1, tipo1, ..., datos)
EXPON.DIST	Devuelve la distribución exponencial. Use EXPON.DIST para modelar el tiempo entre eventos, como cuánto tiempo dedica un cajero automático a entregar efectivo. Por ejemplo, puede usar EXPON.DIST para determinar la probabilidad de que el proceso tarde 1 minuto como mínimo.	**EXPON.DIST**(X, Lambda, Acumulativo)

GENERATE	La expresión de la segunda tabla se evaluará para cada fila de primera tabla. Se devuelve la combinación cruzada de la primera tabla con estos resultados.	**GENERATE**(Table1, Table2)
GENERATEALL	La expresión de la segunda tabla se evaluará para cada fila de primera tabla. Devuelve la combinación cruzada de la primera tabla con estos resultados, incluidas las filas para las que la expresión de la segunda tabla está vacía.	**GENERATEALL**(Table1, Table2)
GEOMEAN	Devuelve la media geométrica de la referencia de columna correspondiente.	**GEOMEAN**(ColumnName)
GEOMEANX	Devuelve la media geométrica de los valores de una expresión en una tabla.	**GEOMEANX**(Tabla, Expresión)

MAXA	Devuelve el valor mayor de una columna. No omite los valores lógicos y el texto.	**MAXA**(ColumnName)
MAXX	Devuelve el valor numérico mayor que se obtiene al evaluar una expresión para cada fila de una tabla.	**MAXX**(Tabla, Expresión)
MEDIAN	Devuelve el percentil 50 de los valores de una columna.	**MEDIAN**(Columna)
MEDIANX	Devuelve el percentil 50 de los valores de una expresión en una tabla.	**MEDIANX**(Tabla, Expresión)
MINA	Devuelve el valor menor de una columna. No omite los valores lógicos y el texto.	**MINA**(ColumnName)
MINX	Devuelve el valor numérico menor que se obtiene al evaluar una expresión para cada fila de una tabla.	**MINX**(Tabla, Expresión)

NORM.DIST	Devuelve la distribución normal para la media y desviación estándar especificadas.	**NORM.DIST**(X, Media, Desv_estándar, Acumulativo)
NORM.INV	Devuelve el inverso de la distribución acumulativa normal para la media y desviación estándar especificadas.	**NORM.INV**(Probabilidad, Media, Desv_estándar)
NORM.S.DIST	Devuelve la distribución normal estándar (tiene una media de cero y una desviación estándar de uno).	**NORM.S.DIST**(Z, Acumulativo)
NORM.S.INV	Devuelve el inverso distribución acumulativa normal estándar. Tiene una media de cero y una desviación estándar de uno.	**NORM.S.INV**(Probabilidad)

PERCENTILE.INC	Devuelve el percentil k (inclusive) de los valores en una columna.	**PERCENTILE.INC**(Columna, K)
PERCENTILEX.EXC	Devuelve el percentil k (exclusive) de los valores de una expresión de una tabla.	**PERCENTILEX.EXC**(Tabla, Expresión, K)
PERCENTILEX.INC	Devuelve el percentil k (inclusive) de los valores de una expresión de una tabla.	**PERCENTILEX.INC**(Tabla, Expresión, K)
POISSON.DIST	Devuelve la distribución de Poisson. Una aplicación común de la distribución de Poisson es predecir el número de eventos en un tiempo específico, como el número de automóviles que llegan a una estación de peaje en 1 minuto.	**POISSON.DIST**(X, Media, Acumulativo)

RANK.EQ	Devuelve la clasificación de un número en una columna de números. Si varios valores tienen la misma clasificación, se devuelve la clasificación superior de dicho conjunto de valores.	**RANK.EQ**(Valor, ColumnName, [Order])
RANKX	Devuelve la clasificación de una expresión evaluada en el contexto actual de la lista de valores para la expresión evaluada por cada fila de la tabla especificada.	**RANKX**(Tabla, Expresión, [Valor], [Order], [Ties])
ROW	Devuelve una tabla de una sola fila con nuevas columnas especificadas por las expresiones DAX.	**ROW**(Nombre1, Expresión1, ...)

SAMPLE	Devuelve un subconjunto de muestra a partir de una expresión de tabla determinada.	**SAMPLE**(Tamaño, Tabla, OrderBy1, [Order1], ...)
SELECTCOLUMNS	Devuelve una tabla con las columnas seleccionadas de la tabla y las columnas	**SELECTCOLUMNS**(Tabla, Nombre1, Expresión1, ...)
	nuevas especificadas por las expresiones DAX.	
SIN	Devuelve el seno del ángulo dado.	**SIN**(Número)
SINH	Devuelve el seno hiperbólico de un número.	**SINH**(Número)

STDEV.P	Calcula la desviación estándar a partir del valor de población total indicado como argumento. Omite los valores lógicos y el texto.	**STDEV.P**(ColumnName)
STDEV.S	Estima la desviación estándar basada en una muestra. Omite los valores lógicos y el texto de la muestra.	**STDEV.S**(ColumnName)
STDEVX.P	Estima la desviación estándar basada en la población total que se obtiene de la evaluación de una expresión por cada fila de una tabla.	**STDEVX.P**(Tabla, Expresión)
STDEVX.S	Estima la desviación estándar basada en una muestra que se obtiene de la evaluación de una expresión por cada fila de una tabla.	**STDEVX.S**(Tabla, Expresión)
SQRTPI	Devuelve la raíz cuadrada de un (número * pi).	**SQRTPI**(Número)

SUMMARIZE	Crea un resumen de la tabla de entrada agrupado por las columnas especificadas.	**SUMMARIZE**(Tabla, [GroupByColumnName1, ..., [Nombre1], [Expresión1], ...)

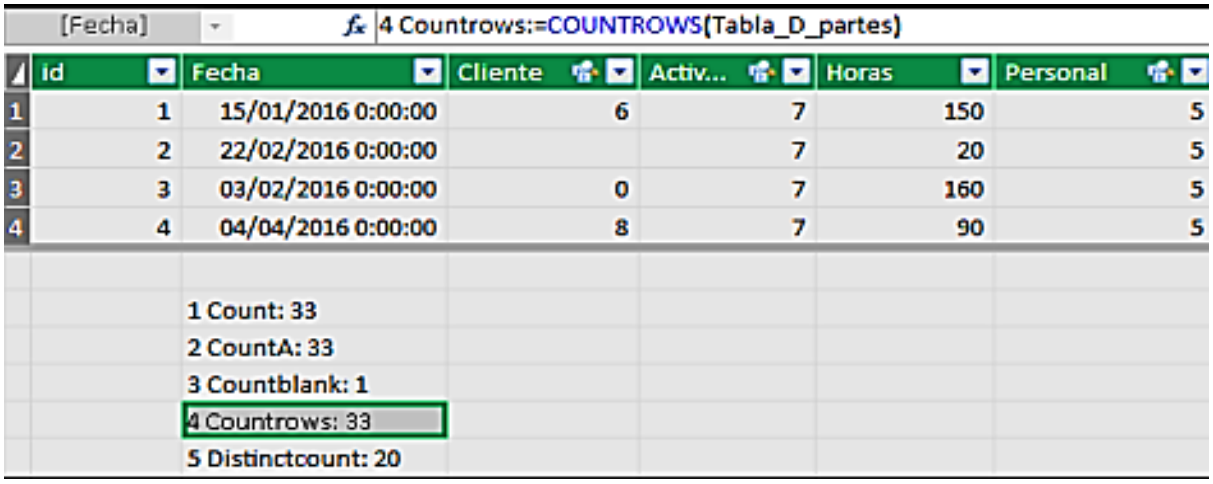
T.DIST	Devuelve la distribución t de cola izquierda de Student.	**T.DIST**(X, Grados_de_libertad)
T.DIST.2T	Devuelve la distribución T bilateral de Student.	**T.DIST.2T**(X, Grados_de_libertad)
T.DIST.RT	Devuelve la distribución t de Student de cola derecha.	**T.DIST.RT**(X, Grados_de_libertad)
T.INV	Devuelve el inverso de cola izquierda de la distribución de Student.	**T.INV**(Probabilidad, Grados_de_libertad)
T.INV.2T	Devuelve el inverso bilateral de la distribución T de Student.	**T.INV.2T**(Probabilidad, Grados_de_libertad)

TAN	Devuelve la tangente del ángulo dado.	**TAN**(Número)
TANH	Devuelve la tangente hiperbólica de un número.	**TANH**(Número)
TOPN	Devuelve el número determinado de filas superiores en función de una expresión especificada.	**TOPN**(N_Value, Tabla, [OrderBy_Expression1], [Order1])

VAR.P	Calcula la varianza según la población total. Omito los valores lógicos y el texto de la población.	**VAR.P**(ColumnName)
VAR.S	Estima la varianza basada en una muestra. Omite los valores lógicos y el texto de la muestra.	**VAR.S**(ColumnName)
VARX.P	Estima la varianza a partir de la población total que se obtiene de la evaluación de una expresión por cada fila de una tabla.	**VARX.P**(Tabla, Expresión)
VARX.S	Estima la varianza basada en una muestra que se obtiene de la evaluación de una expresión por cada fila de una tabla.	**VARX.S**(Tabla, Expresión)

XIRR	Devuelve la tasa interna de devolución para una programación de flujo de efectivo que no es necesariamente periódica.	**XIRR**(Tabla, Valores, Dates, [Guess])
XNPV	Devuelve el valor neto presente para una programación de flujos de efectivo.	**XNPV**(Tabla, Valores, Dates, Rate)

Ejemplo:

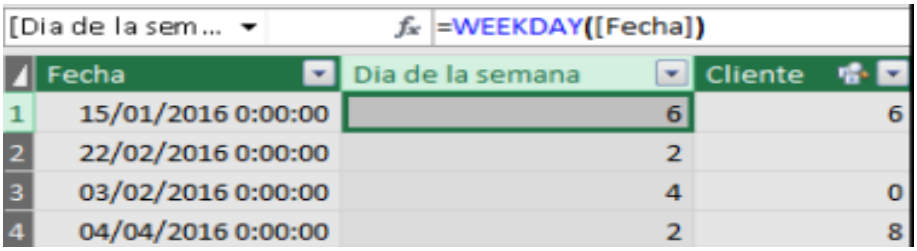


Categoría: Funciones de Fecha y Hora

Funciones más utilizadas:

Función	Descripción	Sintaxis
DATE	Devuelve la fecha especificada en formato de fecha y hora	**DATE**(Año, Mes, Día)
DATEDIFF	Devuelve el número de unidades (unidad especificada en intervalo) entre las dos fechas de entrada	**DATE**(Fecha1, Fecha2, Intervalo)
EDATE	Devuelve la fecha que es el número de meses indicado antes o después de la fecha inicial	**EDATE**(StartDate, Meses)
TODAY	Devuelve la fecha actual en formato de fecha y hora.	**TODAY()**
WEEKDAY	Devuelve un número del 1 al 7 que identifica el día de la semana de una fecha	**WEEKDAY**(Fecha, [ReturnType])
WEEKNUM	Devuelve un número de semana del año	**WEEKNUM**(Fecha, [ReturnType])
YEAR	Devuelve el año de una fecha como número entero de cuatro dígitos	**YEAR**(Fecha)

Ejemplo:



Más funciones en la categoría:

Función	Descripción	Sintaxis
CALENDAR	Devuelve una tabla con una columna de todas les fechas entre StartDate y EndDate.	**CALENDAR**(StartDate, EndDate)
CALENDARAUTO	Devuelve una tabla con una columna de fechas calculadas automáticamente a partir del modelo.	**CALENDARAUTO**([FiscalYearEndMonth])
DATEVALUE	Convierte una fecha en forma de texto a una fecha en formato de fecha y hora.	**DATEVALUE**(DateText)
DAY	Devuelve el número del 1 al 31 que representa el día del mes.	**DAY**(Fecha)
EDATE	Devuelve la fecha que es el número de meses indicado antes o después de la fecha inicial.	**EDATE**(StartDate, Meses)
EOMONTH	Devuelve la fecha, en formato de fecha y hora, del último día del mes antes o después de un número de meses especificado.	**EOMONTH**(StartDate, Meses)
HOUR	Devuelve la hora como un número de 0 (12:00 a.m.) a 23 (11:00 p.m.).	**HOUR**(Datetime)
MINUTE	Devuelve un número del 0 al 59 que representa el minuto.	**MINUTE**(Datetime)
MONTH	Devuelve un número del 1 (enero) al 12 (diciembre) que representa el mes.	**MONTH**(Fecha)
NOW	Devuelve la fecha y la hora actuales en formato de fecha y hora.	**NOW()**
SECOND	Devuelve un número del 0 al 59 que representa el segundo.	**SECOND(Datetime)**
TIME	Convierte las horas, los minutos y los segundos proporcionados como números en una hora en formato de fecha y hora.	**TIME**(Hora, Mínuto, Segundo)
TIMEVALUE	Convierte una hora en formato de texto a una hora en formato de fecha y hora.	**TIMEVALUE**(TimeText)
YEARFRAC	Devuelve la fracción del año que representa el número de días completos entre la fecha_inicial y la fecha_final.	**YEARFRAC**(StartDate, EndDate, [Base])

Categoría: Funciones de Texto

Funciones más utilizadas:

Función	Descripción	Sintaxis
CONCATENATE	Une dos cadenas de texto en una sola	**CONCATENATE**(Text1, Text2)
LEFT	Devuelve el número de caracteres especificados desde el principio de una cadena de texto	**LEFT**(Texto, [NumberOfCharacters])
LEN	Devuelve el número de caracteres de una cadena de texto	**LEFT**(Texto)
RIGHT	Devuelve el número de caracteres especificados desde el final de una cadena de texto	**RIGHT**(Texto, [NumberOfCharacters])
SEARCH	Devuelve la primera posición de una cadena de texto en otra. SEARCH no distingue mayúsculas de mayúsculas.	**SEARCH**(FindText, WithinText, [StartPosition], [NotFoundValue])
SUBSTITUTE	Reemplaza texto existente por otro nuevo en una cadena de texto	**SUBSTITUTE**(Texto, OldText, NewText, [InstanceNumber])

Más funciones en la categoría:

Función	Descripción	Sintaxis
BLANK	Devuelve un espacio en blanco.	**BLANK()**
CODE		
CONCATENATEX	Evalúa la expresión para cada fila de la tabla, a concatenación de esos valores en un resultado de	**CONCATENATEX**(Tabla, Expresión, [Delimitador],
	cadena única, separados por el delimitador especificado.	[OrderBy_Expression1, [Order1], ...)
EXACT	Comprueba si dos cadenas de texto son exactamente iguales y devuelve VERDADERO o FALSO. IGUAL distingue mayúsculas de minúsculas.	**EXACT**(Text1, Text2)

FIND	Devuelve la primera posición de una cadena de texto en otra. ENCONTRAR distingue mayúsculas de minúsculas.	**FIND**(FindText, WithinText, [StartPosition], [NotFoundValue])
FIXED	Redondea un número de decimales especificado y devuelve el resultado como texto con comas opcionales.	**FIXED**(Número, [Decimals], [NoCommas])
FORMAT	Convierte un valor en texto con el formato de número especificado.	**FORMAT**(Valor, Formato)
LOWER	Convierte todas las letras de una cadena de texto a minúsculas.	**LOWER**(Texto)
MID	Devuelve una cadena de caracteres desde la mitad de una cadena de texto, para una posición inicial y una longitud dadas.	**MID**(Texto, StartPosition, NumberOfCharacters)

REPLACE	Reemplaza parte de una cadena de texto por otra distinta	**REPLACE**(OldText, StartPosition, NumberOfCharacters, NewText)
REPT	Repite un texto un determinado número de veces. Use REPT para llenar una celda con un número de instancias cadena de texto.	**REPT**(Texto, NumberOfTimes)

TRIM	Quita todos los espacios de una cadena de texto, excepto los espacios individuales entre palabras.	**TRIM**(Texto)
UNICHAR	Devuelve el carácter Unicode al que hace referencia el valor numérico correspondiente.	**UNICHAR**(Número)
UPPER	Convierte una cadena de texto a letras mayúsculas.	**UPPER**(Texto)
VALUE	Convierte en un número una cadena de texto que representa un número.	**VALUE**(Texto)

Categoría: Funciones de Filtrado

Funciones más utilizadas:

Función	Descripción	Sintaxis
ALL	Devuelve todas las filas de una tabla, o todos los valores de una columna, omitiendo los filtros que se hayan aplicado	**ALL**(TableNameOrColumnName, [ColumnName1], ...)
CALCULATE	Evalúa una expresión en un contexto modificado por filtros	**CALCULATE**(Expresión, [Filtrar1], ...)
FILTER	Devuelve una tabla que se ha filtrado	**FILTER**(Tabla, FilterExpression)
RELATED	Devuelve un valor relacionado de otra tabla	**RELATED**(ColumnName)

Más funciones en la categoría:

Función	Descripción	Sintaxis
ADDMISSINGITEMS	Vuelve a agregar las filas con valores de medidas vacíos	**ADDMISSINGITEMS**([ShowAll_ColumnName1], …, Tabla, [GroupBy_ColumnName1], …, [FilterTable1], …)
ALLEXCEPT	Devuelve todas las filas de una tabla excepto las filas afectadas por los filtros de columna especificados.	**ALLEXCEPT**(TableName, ColumnName1, …)
ALLNOBLANKROW	Devuelve todas las filas excepto las filas en blanco de una tabla, o todos los valores de una columna, omitiendo los filtros que se pueden haber aplicado.	**ALLNOBLANKROW**(TableNameOrColumnName, [ColumnName1], …)
ALLSELECTED	Devuelve todas las filas de una tabla, o todos los valores de una columna omitiendo los filtros que se hayan aplicado dentro de la consulta, pero conservando los filtros externos.	**ALLSELECTED**([TableNameOrColumnName])

CALCULATETABLE	Evalúa una expresión de tabla en un contexto modificado por filtros.	**CALCULATETABLE**(Tabla, [Filtrar1], …)
CROSSFILTER	Especifica la dirección del filtrado cruzado que se va a usar en la evaluación de una expresión DAX. La relación se define con los nombre, como argumentos, de las columnas que sirven de extremos.	**CROSSFILTER**(LeftColumnName, RightColumnName, CrossFilterType)
DISTINCT	Devuelve una tabla con una sola columna que contiene valores distintos (únicos) en una columna, para un argumento de columna. O bien, varias columnas con una combinación distinta (única) de valores, para un argumento de expresión de tabla.	**DISTINCT**(ColumnNameOrTableExpr)

EARLIER	Devuelve el valor de la columna anterior al número especificado de recorridos de tabla (el valor predeterminado es 1)	**EARLIER**(ColumnName, [Número])
EARLIEST	Devuelve el valor de la columna para el primer punto en el que había un contexto de fila.	**EARLIEST**(ColumnName)

HASONEFILTER	Devuelve True si la tabla o la columna especificada tiene exactamente un filtro.	**HASONEFILTER**(ColumnName)
HASONEVALUE	Devuelve True si solo hay un valor en la columna especificada.	**HASONEVALUE**(ColumnName)

ISCROSSFILTERED	Devuelve True si la tabla o la columna especificada tiene un filtro cruzado.	**ISCROSSFILTERED**(TableNameOrColumnName)
ISFILTERED	Devuelve True si hay filtros directos en la columna especificada.	**ISFILTERED**(ColumnName)

KEEPFILTERS	Cambia la semántica de filtrado de la función CALCULATE y CALCULATETABLE.	**KEEPFILTERS**(Expresión)
RELATEDTABLE	Devuelve las tablas relacionadas filtradas de forma que solo se incluyan las filas relacionadas.	**RELATEDTABLE**(Tabla)

SELECTEDVALUE	Devuelve el valor cuando solo hay un valor en la columna especificada; de lo contrario, devuelve el resultado alternativo.	**SELECTEDVALUE**(Column Name, [AlternateResult])

SUBSTITUTEWITHINDEX	Devuelve una tabla que representa la semicombinación de dos tablas proporcionadas y para la que el conjunto común de columnas se	**SUBSTITUTEWITHINDEX**(T abla, Nombre,
	reemplaza por una columna de índice de base 0. El índice se basa en las filas de la segunda tabla ordenadas por las expresiones de orden especificadas.	SemiJoinIndexTable, Expresión1, [Order1], ...)
USERELATIONSHIP	Especifica una relación existente para usarla en la evaluación de una expresión DAX. La relación se define agregando un nombre, igual que los argumentos, a las dos columnas que sirven de extremos.	**USERELATIONSHIP**(Colum nName1, ColumnName2)
VALUES	Devuelve una tabla con una sola columna o una tabla que contiene los valores distintivos (únicos) de una columna.	**VALUES**(TableNameOrColu mnName)

Categoría: Funciones de Inteligencia de Tiempo

Funciones más utilizadas:

Función	Descripción	Sintaxis
DATEADD	Desplaza el conjunto de fechas indicado en un intervalo especificado.	**DATEADD**(Dates, NumberOfIntervals, Interval)
DATESINPERIOD	Devuelve las fechas del periodo especificado	**DATESINPERIOD**(Dates, StartDate, NumberOfIntervals, Interval)
DATESYTD	Devuelve un conjunto de fechas del año hasta la fecha actual	**DATESYTD**(Dates, [YearEndDate])
TOTALYTD	Evalúa la expresión especificada a lo largo del intervalo que empieza el primer día del año y termina con la última fecha de la columna de fecha especificada después de aplicar los filtros especificados	**TOTALYTD**(Expresión, Dates, [Filtrar], [YearEndDate])

Más funciones en la categoría:

Función	Descripción	Sintaxis
CLOSINGBALANCEMONTH	Evalúa de la expresión especificada para la fecha correspondiente al final del mes actual después de aplicar los filtros especificados.	**CLOSINGBALANCEMONTH**(Expresión, Dates, [Filtrar])
CLOSINGBALANCEQUARTER	Evalúa de la expresión especificada para la fecha correspondiente al final del trimestre actual después de aplicar los filtros especificados.	**CLOSINGBALANCEQUARTER**(Expresión, Dates, [Filtrar])
CLOSINGBALANCEYEAR	Evalúa de la expresión especificada para la fecha correspondiente al final del año actual después de aplicar los filtros especificados.	**CLOSINGBALANCEYEAR**(Expresión, Dates, [Filtrar], [YearEndDate])

DATESBETWEEN	Devuelve las fechas entre dos fechas dadas.	**DATESBETWEEN**(Dates, StartDate, EndDate)
DATESMTD	Devuelve un conjunto de fechas del mes hasta la fecha actual.	**DATESMTD**(Dates)
DATESQTD	Devuelve un conjunto de fechas del trimestre hasta la fecha actual.	**DATESQTD**(Dates)
ENDOFMONTH	Devuelve el final del mes.	**ENDOFMONTH**(Dates)
ENDOFQUARTER	Devuelve el final del trimestre.	**ENDOFQUARTER**(Dates)
ENDOFYEAR	Devuelve el final del año.	**ENDOFYEAR**(Dates, [YearEndDate])

FIRSTDATE	Devuelve la primera fecha que no está en blanco.	**FIRSTDATE**(Dates)
FIRSTNONBLANK	Devuelve el primer valor de la columna para el que la expresión tiene un valor que no está en blanco.	**FIRSTNONBLANK**(ColumnName, Expresión)

LASTDATE	Devuelve la última fecha que no está en blanco.	**LASTDATE**(Dates)
LASTNONBLANK	Devuelve el último valor de la columna para el que la expresión tiene un valor que no está en blanco.	**LASTNONBLANK**(ColumnName, Expresión)

NEXTDAY	Devuelve el día siguiente.	**NEXTDAY**(Dates)
NEXTMONTH	Devuelve el mes siguiente.	**NEXTMONTH**(Dates)
NEXTQUARTER	Devuelve el trimestre siguiente.	**NEXTQUARTER**(Dates, [YearEndDate])
NEXTYEAR	Devuelve el año siguiente.	**NEXTYEAR**(Dates, [YearEndDate])

OPENINGBALANCEMONTH	Evalúa la expresión especificada para la fecha correspondiente al final del mes anterior después de aplicar los filtros especificados.	**OPENINGBALANCEMONTH**(Expresión, Dates, [Filtrar])
OPENINGBALANCEQUARTER	Evalúa la expresión especificada para la fecha correspondiente al final del trimestre anterior después de aplicar los filtros especificados.	**OPENINGBALANCEQUARTER**(Expresión, Dates, [Filtrar])
OPENINGBALANCEYEAR	Evalúa la expresión especificada para la fecha correspondiente al final del año anterior después de aplicar los filtros especificados.	**OPENINGBALANCEYEAR**(Expresión, Dates, [Filtrar], [YearEndDate])
PARALLELPERIOD	Devuelve un período paralelo de fechas según el conjunto de fechas dado y un intervalo especificado.	**PARALLELPERIOD**(Dates, NumberOfIntervals, Interval)
PREVIOUSYEAR	Devuelve el año anterior.	**PREVIOUSYEAR**(Dates, [YearEndDate])
SAMEPERIODLASTYEAR	Devuelve un conjunto de fechas en la selección actual del año anterior.	**SAMEPERIODLASTYEAR**(Dates)
STARTOFMONTH	Devuelve el inicio del mes.	**STARTOFMONTH**(Dates)
STARTOFQUARTER	Devuelve el inicio del trimestre.	**STARTOFQUARTER**(Dates)
STARTOFYEAR	Devuelve el inicio del año.	**STARTOFYEAR**(Dates, [YearEndDate])
TOTALMTD	Evalúa expresión especificada a lo largo del intervalo que empieza al principio del mes y termina con la última fecha de la columna de fecha especificada después de aplicar los filtros especificados.	**TOTALMTD**(Expresión, Dates, [Filtrar])
TOTALQTD	Evalúa la expresión especificada a lo largo del intervalo que empieza el primer día del trimestre y termina con la última fecha de la columna de fecha especificada después de aplicar los filtros especificados.	**TOTALQTD**(Expresión, Dates, [Filtrar])

Categoría: Funciones de Información

Funciones más utilizadas:

Función	Descripción	Sintaxis
ISBLANK	Comprueba sin un valor está en blanco y devuelve TRUE o FALSE	**ISBLANK**(Valor)
ISERROR	Comprueba sin un valor es un error y devuelve TRUE o FALSE	**ISERROR**(Valor)
ISNUMBER	Comprueba sin un valor es un número y devuelve TRUE o FALSE	**ISNUMBER**(Valor)
ISTEXT	Comprueba sin un valor es un texto y devuelve TRUE o FALSE	**ISTEXT**(Valor)

Más funciones en la categoría:

Función	Descripción	Sintaxis
CONTAINS	Devuelve TRUE si hay al menos una fila con valores especificados en todas las columnas.	**CONTAINS**(Tabla, ColumnName1, Valor1, ...)
CUSTOMDATA	Devuelve el valor de la propiedad CustomData de la cadena de conexión, si está definida; en caso contrario BLANK().	**CUSTOMDATA()**
ISEVEN	Devuelve TRUE si el número es par o FALSE si el número es impar.	**ISEVEN**(Número)
ISINSCOPE	Devuelve un valor "true" si la columna especificada es el nivel de una jerarquía de niveles.	**ISINSCOPE(ColumnName)**
ISLOGICAL	Comprueba si un valor es lógico (TRUE o FALSE) y devuelve TRUE o FALSE.	**ISLOGICAL**(Valor)
ISNONTEXT	Comprueba si un valor no es de texto (las celdas en blanco no son texto) y devuelve TRUE o FALSE.	**ISNONTEXT**(Valor)
ISONORAFTER	IsOnOrAfter es una función booleana que emula el comportamiento de la cláusula Start At y devuelve true para una fila que cumple todas las condiciones mencionadas como parámetros en esta función.	**ISONORAFTER**(Value11, Value21, [Order1], ...)
LOOKUPVALUE	Recupera un valor de una tabla.	**LOOKUPVALUE**(Result_ColumnName, Search_ColumnName1, Search_Value1)
USERNAME	Devuelve el nombre de dominio y el nombre de usuario de la conexión actual, con el formato: domino\nombre de usuario.	**USERNAME()**

Categoría: Funciones Primarias y Secundarias

Funciones de la categoría:

Función	Descripción	Sintaxis
PATH	Devuelve una cadena que contiene una lista delimitada de identificadores, empezando por la parte superior o raíz de una jerarquía y terminando con el identificador especificado	**PATH**(ID_ColumnName, Parent_ColumnName)
PATHCONTAINS	Devuelve TRUE si el elemento especificado existe en la ruta de acceso especificada	**PATHCONTAINS**(Ruta de acceso, Elemento)
PATHITEM	Devuelve el enésimo elemento de la lista delimitada creada por la función Ruta	**PATHITEM**(Ruta de acceso, Posición, [Tipo])
PATHITEMREVERSE	Devuelve el enésimo elemento de la lista delimitada creada por la función Ruta, contando hacia atrás desde el último elemento de la ruta de acceso	**PATHITEMREVERSE**(Ruta de acceso, Posición, [Tipo])
PATHLENGTH	Devuelve el número de elementos de una cadena de ruta de acceso determinada. Esta función devuelve 1 para la ruta generada para un identificador en la parte superior o raíz de una jerarquía.	**PATHLENGTH**(Ruta de acceso)

Categoría: Otras Funciones

Funciones más utilizadas:

Función	Descripción	Sintaxis
DATATABLE	Devuelve una tabla con datos definidos en línea	**DATATABLE**(nombre1, tipo1, ..., datos)
GROUPBY	Crea un resumen de la tabla de entrada agrupado por las columnas especificadas	**GROUPBY**(Tabla, [GroupBy_ColumnName1], ..., [Nombre1], [Expresión1], ...)

Más funciones en la categoría:

Función	Descripción	Sintaxis
ERROR	Genera un error especificado por el usuario.	**ERROR**(ErrorText)
EXCEPT	Devuelve las filas de la tabla del lado izquierdo que no aparece en la tabla de la derecha.	**EXCEPT**(LeftTable, RightTable)
GENERATESERIES	Devuelve una tabla con una columna, rellena con valores	**GENERATESERIES**(StartValue, EndValue, [IncrementValue])
	secuenciales de principio a fin.	
INTERSECT	Devuelve las filas de la tabla de la izquierda que aparecen en la tabla de la derecha.	**INTERSECT**(LeftTable, RightTable)
ISEMPTY	Devuelve true si la tabla especificada o la expresión de tabla es Empty.	**ISEMPTY**(Tabla)
NATURALINNERJOIN	Combina la tabla izquierda con la tabla derecha usando semántica de combinación interna.	**NATURALINNERJOIN**(LeftTable, RightTable)
NATURALLEFTOUTERJOIN	Une la tabla izquierda con la tabla derecha usando la semántica de combinación externa izquierda.	**NATURALLEFTOUTERJOIN**(LeftTable, RightTable)
SUMMARIZECOLUMNS	Crea una tabla de resumen para los totales solicitados de un conjunto de grupos.	**SUMMARIZECOLUMNS**([GroupBy_ColumnName1], ..., [FilterTable1], ..., [Nombre1], [Expresión1], ...)
TREATAS	Trata las columnas de la tabla de entrada como columnas de otras tablas. Para cada columna, se filtran los valores que no figuran en su respectiva columna de salida.	**TREATAS**(Expresión, ColumnName1, ...)
UNION	Devuelve la unión de las dos tablas cuyas columnas coinciden.	**UNION**(Tabla1, ...)

CAPÍTULO CUATRO

CREAR Y ADMINISTRAR RELACIONES EN POWER BI

Introducción

Según (Agreda-Siguenza, s.f.):

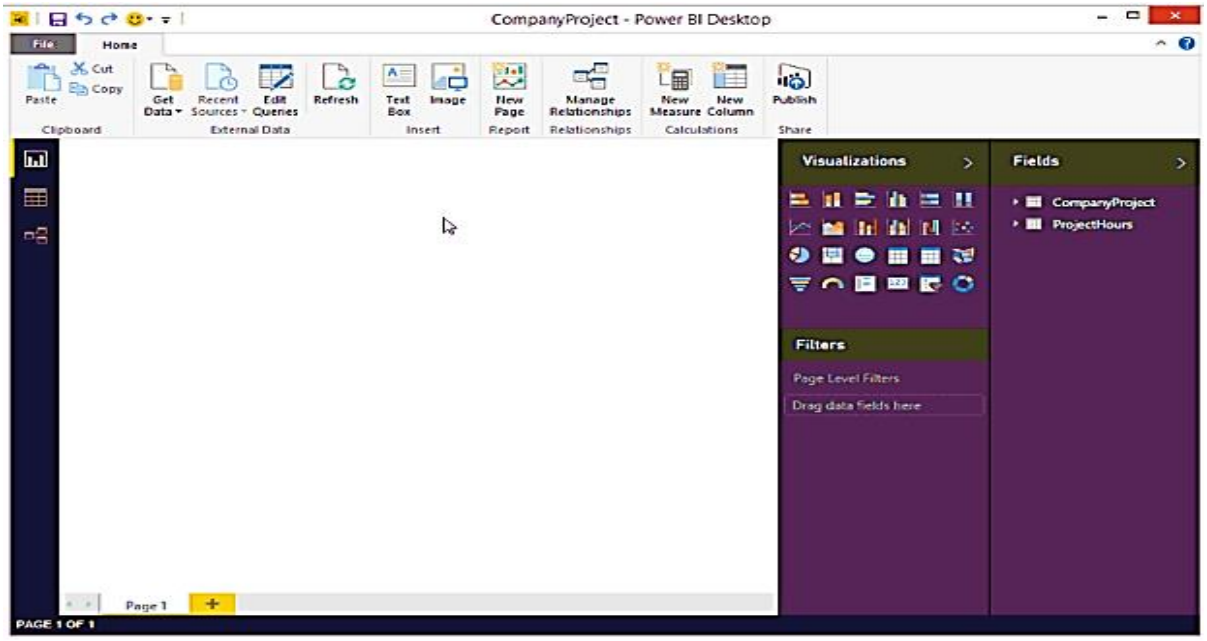
Al tener varias tablas, lo más probable es que se vaya a realizar un análisis con los datos de todas ellas. Las relaciones entre esas tablas son necesarias para calcular los resultados de forma precisa y mostrar la información correcta en los informes. En la mayoría de los casos no tendrás que hacer nada. La característica de detección automática lo hace automáticamente. Aunque es posible que, en ocasiones, tenga que crear relaciones o realizar cambios en una relación. En cualquier caso, es importante entender las relaciones en Power BI Desktop y cómo crearlas y editarlas.

Detección automática durante la carga

Si consulta dos o más tablas al mismo tiempo, cuando se carguen los datos, Power BI Desktop intenta buscar y crear relaciones automáticamente. Las opciones de relación Cardinalidad, Dirección de filtro cruzado y Activar esta relación se establecen automáticamente. Power BI Desktop examina los nombres de columna en las tablas que se consultan para determinar si hay posibles relaciones. Si las hay, esas relaciones se crean automáticamente. Si Power BI Desktop no puede determinar con un alto nivel de confianza que hay una coincidencia, no crea automáticamente la relación. Sin embargo, puede usar el cuadro de diálogo Administrar relaciones para crear o modificar manualmente las relaciones.

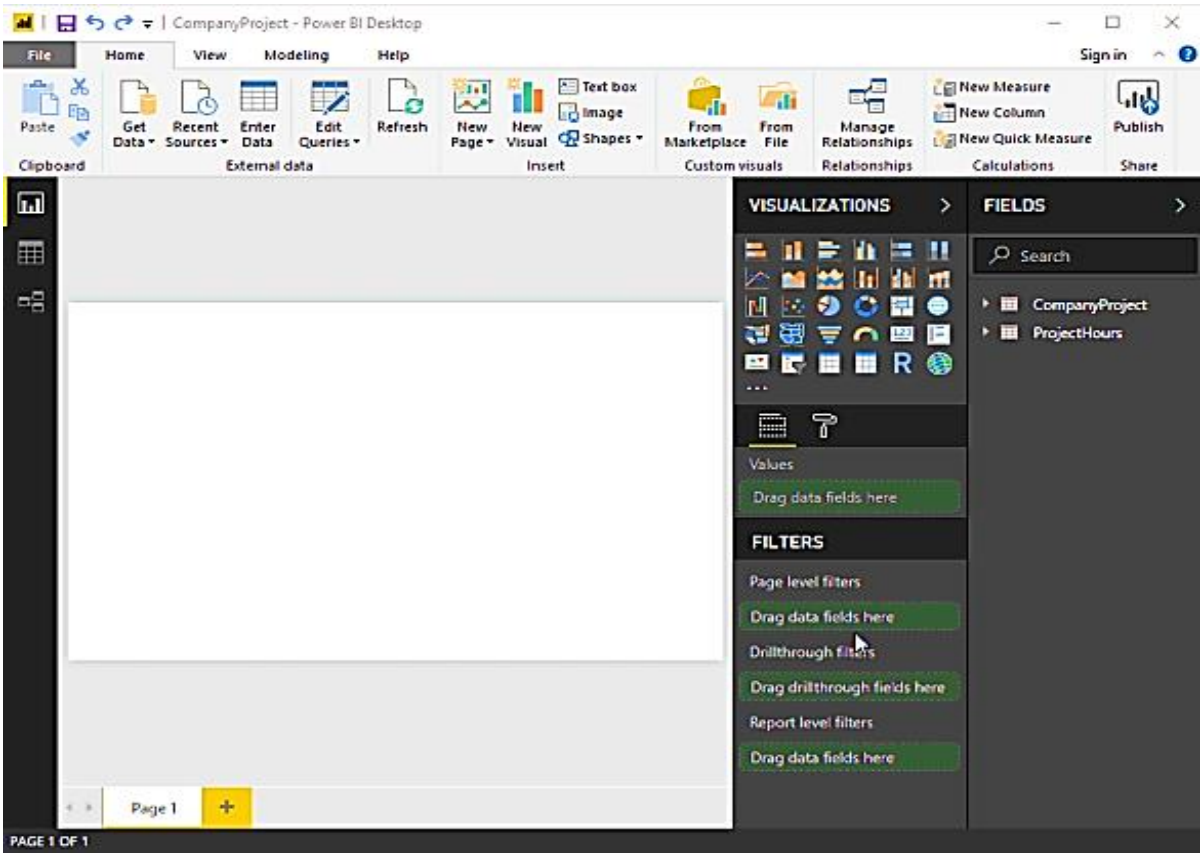
Creación de una relación con detección automática

En la pestaña Modelado, haga clic en Administrar relaciones>Detección automática.



Crear una relación de forma manual

1. En la pestaña Modelado, haga clic en Administrar relaciones>Nuevo.
2. En el cuadro de diálogo Crear relación, en la primera lista desplegable de tablas, seleccione una tabla. Seleccione la columna que quiera usar en la relación.
3. En la segunda lista desplegable de tablas, seleccione la otra tabla que quiera incluir en la relación. Seleccione la otra columna que quiera usar y seleccione Aceptar.



De forma predeterminada, Power BI Desktop configura automáticamente las opciones Cardinalidad (dirección), Dirección de filtro cruzado y Activar esta relación en la nueva relación. Sin embargo, puede cambiar esta configuración si es necesario.

Si ninguna de las tablas seleccionadas para la relación tiene valores únicos, verá el siguiente error: Una de las columnas debe tener valores únicos. Al menos una tabla de una relación debe tener una lista distinta y única de valores de clave, que es un requisito común para todas las tecnologías de bases de datos relacionales.

Si detecta ese error, hay un par de formas de corregir el problema:

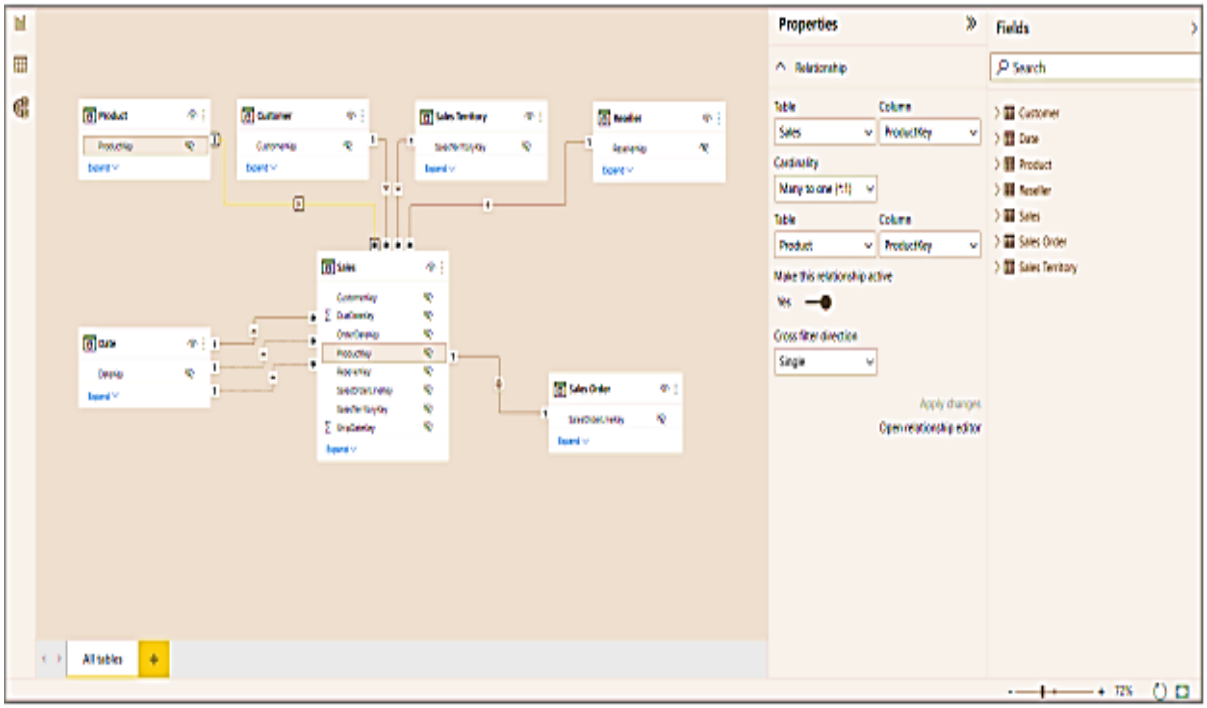
- Usar Quitar filas duplicadas para crear una columna con valores únicos. El inconveniente de este enfoque es que podría perder información al quitar filas duplicadas. A menudo, una clave (fila) se duplica por un buen motivo.
- Agregar una tabla intermedia hecha de la lista de valores de claves distintos en el modelo, que luego se vinculará a ambas columnas originales de la relación.

Como alternativa, en los diseños del diagrama de vista Modelo, puede arrastrar y colocar una columna de una tabla a una columna de otra tabla para crear una relación.

Editar una relación

Hay dos maneras de editar una relación en Power BI.

El primer método para editar una relación es usar las relaciones de edición en el panel Propiedades de la vista Modelo, donde puede seleccionar cualquier línea entre dos tablas para ver las opciones de relación en el panel Propiedades. Asegúrese de expandir el panel Propiedades para ver las opciones de relación.



Propiedades

El otro método de edición de una relación es usar el cuadro de diálogo Editor de relaciones, que puede abrir muchas maneras desde dentro de Power BI Desktop. En la lista siguiente se muestran diferentes formas de abrir el cuadro de diálogo Editor de relaciones:

En la vista Informe, realice cualquiera de las acciones siguientes:

- Seleccione la cinta Modelado, >Administrar relaciones y, a continuación, seleccione la relación y seleccione Editar.

- Seleccione una tabla en la lista Campos y, a continuación, seleccione la cinta Herramientas de tabla, >Administrar relaciones, luego seleccione la relación y a continuación seleccione Editar.

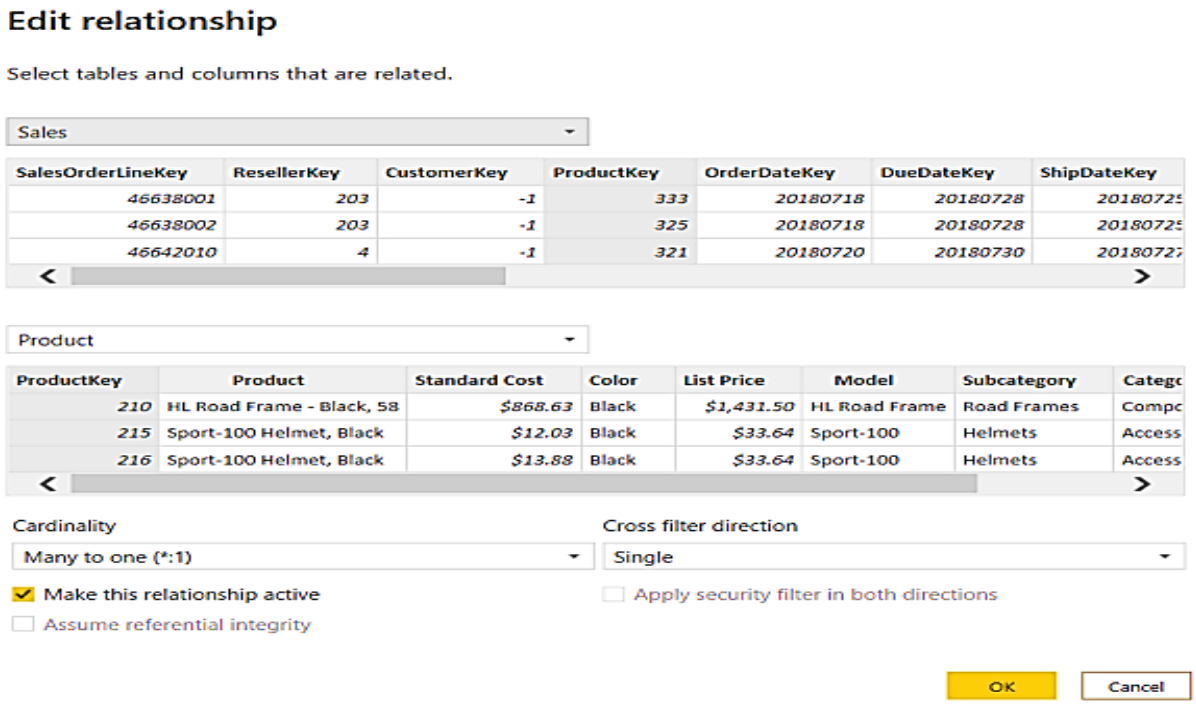
En la vista Datos, seleccione la cinta Herramientas de tabla, >Administrar relaciones, luego seleccione la relación y, a continuación, elija Editar.

En la vista Modelo, realice cualquiera de las acciones siguientes:

- Seleccione la cinta Inicio, >Administrar relaciones, luego elija la relación y, a continuación, seleccione Editar.
- Haga doble clic en cualquier línea entre dos tablas.
- Haga clic con el botón derecho en cualquier línea entre dos tablas y, a continuación, elija Propiedades.
- Seleccione cualquier línea entre dos tablas y, a continuación, elija Abrir editor de relaciones en el panel Propiedades.

Por último, también puede editar una relación desde cualquier vista, hacer clic con el botón derecho o seleccionar los puntos suspensivos para llegar al menú contextual de cualquier tabla, seleccionar Administrar relaciones, seleccionar la relación y, después, Editar

En la imagen siguiente se muestra una captura de pantalla de la ventana Editar relación.



Edición de relaciones mediante diferentes métodos

Utilizar el cuadro de diálogo Editar relaciones es una experiencia más guiada para editar relaciones en Power BI y actualmente está en versión preliminar. Puede ver una vista previa de los datos en cada tabla. A medida que selecciona columnas diferentes, la ventana valida automáticamente la relación y ofrece las selecciones de cardinalidad y filtro cruzado adecuadas.

La edición de relaciones en el panel Propiedades es un enfoque simplificado para editar relaciones en Power BI. Solo verá los nombres y columnas de tabla desde las que puede elegir, no se le presentará una vista previa de datos y las opciones de relación que realice solo se validan al seleccionar Aplicar cambios. El uso del panel Propiedades y su enfoque simplificado reduce el número de consultas generadas al editar una relación, lo que puede ser importante para escenarios de macrodatos, especialmente cuando se usan conexiones de DirectQuery. Las relaciones creadas con el panel Propiedades también pueden ser más avanzadas que las relaciones permitidas para crearse en el cuadro de diálogo Editar relaciones.

También puede seleccionar varias relaciones en los diseños del diagrama de vista Modelo presionando la tecla Ctrl y seleccionando más de una línea para elegir varias relaciones. Las propiedades comunes se pueden editar en el panel Propiedades y Aplicar cambios procesarán los cambios en una transacción.

También se pueden eliminar relaciones únicas o seleccionadas presionando Eliminar en el teclado. No se puede deshacer la acción de eliminación, por lo que un cuadro de diálogo le pide que confirme la eliminación de las relaciones.

Configuración de más opciones

Al crear o editar una relación, puede configurar más opciones. De forma predeterminada, Power BI Desktop configura automáticamente más opciones en función de una mejor aproximación, que puede ser diferente para cada relación según los datos de las columnas.

Cardinalidad

La opción Cardinalidad puede tener una de las siguientes opciones:

Varios a uno (*:1): una relación de varios a uno es el tipo más común de relación predeterminada. Significa que la columna de una tabla concreta puede tener más de una instancia de un valor y que la otra tabla relacionada, a menudo conocida como tabla de búsqueda, solo tiene una instancia de un valor.

Uno a uno (1:1): en una relación uno a uno, la columna de una tabla solo tiene una instancia de un valor determinado y la otra tabla relacionada solo tiene una instancia de un valor determinado.

Uno a varios (1:*): en una relación uno a varios, la columna de una tabla solo tiene una instancia de un valor y la otra tabla relacionada puede tener más de una instancia de un valor.

Varios a varios (*:*): con los modelos compuestos, puede establecer relaciones de varios a varios entre tablas, lo que elimina los requisitos de los valores únicos de las tablas.

También permite descartar las soluciones alternativas anteriores, como el hecho de presentar nuevas tablas solo para establecer relaciones. Para obtener más información, consulte Relaciones con una cardinalidad de varios a varios.

Dirección de filtro cruzado

La opción Dirección de filtro cruzado puede tener una de las siguientes opciones:

Ambos: indica que, a efectos de filtrado, ambas tablas se tratan como si fueran una sola tabla. La opción Ambos funciona bien con una única tabla que tenga muchas tablas de búsqueda a su alrededor. Un ejemplo es una tabla de datos reales de ventas con una tabla de búsqueda del departamento. Esta configuración suele denominarse configuración de esquema de estrella (una tabla central con varias tablas de búsqueda).

Aunque, si tiene dos o más tablas que también tengan tablas de búsqueda (con algo en común), no querrá usar la opción Ambos. Para continuar con el ejemplo anterior, en este caso, también dispone de una tabla de ventas de presupuesto que registra el presupuesto de destino para cada departamento. Además, la tabla de departamento está conectada a la tabla de presupuesto y de ventas. Evite la opción Ambos para este tipo de configuración.

Único: dirección predeterminada más común, que significa que las opciones de filtrado en tablas conectadas trabajan sobre la tabla donde se agregan los valores. Si importa un modelo de datos de Power Pivot o anterior en Excel 2013, todas las relaciones tendrán una dirección única.

Activar esta relación

Si se activa, significa que la relación actúa como la relación predeterminada y activa. En casos donde hay más de una relación entre dos tablas, la relación activa proporciona una manera para que Power BI Desktop cree automáticamente visualizaciones que incluyan las dos tablas.

Descripción de las relaciones

Una vez ha conectado juntas dos tablas con una relación, puede trabajar con los datos en ambas tablas como si fueran una sola tabla. A continuación, no tiene que preocuparse por los detalles de la relación o aplanar esas tablas en una sola tabla antes de importarlas. En muchas situaciones, Power BI Desktop puede crear automáticamente relaciones por usted. Sin embargo, si Power BI Desktop no puede determinar con un grado alto de certeza que deba existir una relación entre dos tablas, no creará la relación automáticamente. En ese caso, deberá hacerlo usted.

Vamos a hacer un breve tutorial para mostrarle mejor cómo funcionan las relaciones en Power BI Desktop.

Esta primera tabla, ProjectHours, es un registro de los vales de trabajo que registran el número de horas que una persona ha trabajado en determinado proyecto.

ProjectHours

Ticket	SubmittedBy	Hours	Project	DateSubmit
1001	Brewer, Alan	22	Azul	1/1/2013
1002	Brewer, Alan	26	Rojo	2/1/2013
1003	Ito, Shu	34	Amarillo	12/4/2012
1004	Brewer, Alan	13	Naranja	1/2/2012
1005	Bowen, Eli	29	Púrpura	1/10/2013
1006	Bento, Nuno	35	Verde	2/1/2013
1007	Hamilton, David	10	Amarillo	1/10/2013
1008	Han, Mu	28	Naranja	1/2/2012
1009	Ito, Shu	22	Púrpura	2/1/2013
1010	Bowen, Eli	28	Verde	10/1/2013
1011	Bowen, Eli	9	Azul	10/15/2013

Esta segunda tabla, CompanyProject, es una lista de proyectos con una prioridad asignada: A, B o C.

CompanyProject

ProjName	Priority
Azul	A
Rojo	B
Verde	C
Amarillo	C
Púrpura	B
Naranja	C

Observe que cada tabla tenga una columna de proyecto. Cada una tiene un nombre ligeramente distinto, pero los valores parecen iguales. Esta diferencia es importante y volveremos a abordarla en breve.

Ahora que tenemos nuestras dos tablas importadas en un modelo, vamos a crear un informe. Lo primero que queremos obtener es el número de horas presentadas por prioridad del proyecto, así que seleccionamos Priority y Hours en el panel Campos.



Si observamos nuestra tabla en el lienzo del informe, verá que el número de horas es de 256 para cada proyecto, que es también el total. Claramente este número no es correcto. ¿Por qué? Esto se debe a que no se puede calcular una suma total de valores de una tabla (Hours en la tabla Project), segmentada por valores en otra tabla (Priority en la tabla CompanyProject) sin establecer una relación entre estas dos tablas.

Por lo tanto, vamos a crear una relación entre estas dos tablas.

¿Recuerda las columnas que vimos en ambas tablas, con un nombre de proyecto, pero con valores similares? Utilizaremos estas dos columnas para crear una relación entre las tablas.

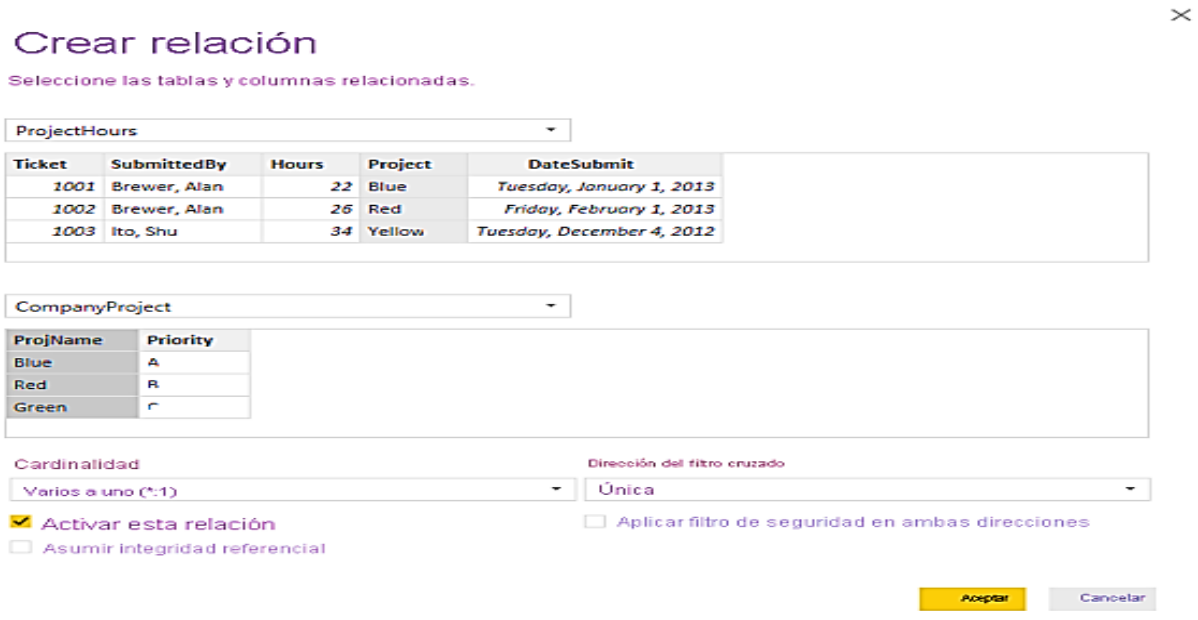
¿Por qué estas columnas? Bueno, si miramos la columna Project de la tabla ProjectHours, veremos valores como Azul, Rojo, Amarillo, Naranja, etc. De hecho, veremos varias filas que tienen el mismo valor. En efecto, tenemos muchos valores de color para Project.

Si miramos la columna ProjName de la tabla CompanyProject, veremos que solo hay uno de cada uno de los valores de color para el nombre de proyecto. El valor de cada color en esta tabla es único y eso es importante, porque podemos crear una relación entre estas dos tablas. En este caso, una relación varios a uno. En una relación de varios a uno, al menos

una columna en una de las tablas debe contener valores únicos. Hay algunas opciones más para ciertas relaciones, que veremos más adelante. Por ahora, vamos a crear una relación entre las columnas del proyecto en cada una de nuestras dos tablas.

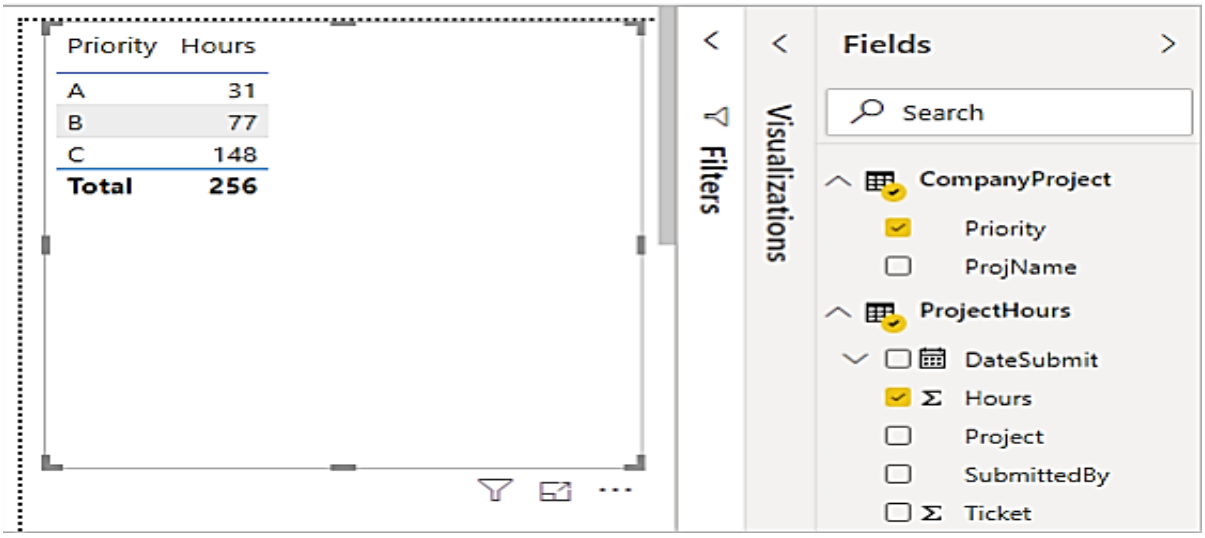
Para crear la nueva relación

1. En la pestaña Modelado, haga clic en Administrar relaciones.
2. En Administrar relaciones, seleccione Nuevo para abrir el cuadro de diálogo Crear relación, donde se pueden seleccionar las tablas, las columnas y cualquier otra opción que queramos para nuestra relación.
3. En la primera lista desplegable, seleccione ProjectHours como primera tabla y, luego, elija la columna Project. Se trata del lado varios de nuestra relación.
4. En la segunda lista desplegable, CompanyProject está preseleccionada como segunda tabla. Seleccione la columna ProjName. Se trata del lado uno de nuestra relación.
5. Conserve los valores predeterminados para las opciones de relación y seleccione Aceptar.
6. En el cuadro de diálogo Administrar relaciones, seleccione Cerrar.



En aras de una divulgación completa, acaba de crear esta relación por las malas. Podría haber seleccionado Detección automática en el cuadro de diálogo Administrar relaciones. De hecho, la detección automática habría creado automáticamente la relación por usted al cargar los datos si las dos columnas tuvieran el mismo nombre.

Ahora, echemos un vistazo en la tabla del lienzo del informe una vez más.



Cuando se suman horas por Priority, Power BI Desktop busca todas las instancias de los valores de color únicos en la tabla de búsqueda CompanyProject, busca todas las instancias de cada uno de esos valores en la tabla ProjectHours y calcula una suma total para cada valor único.

Con la detección automática, puede que ni siquiera tuviera que hacer tanto.

Descripción de las opciones adicionales

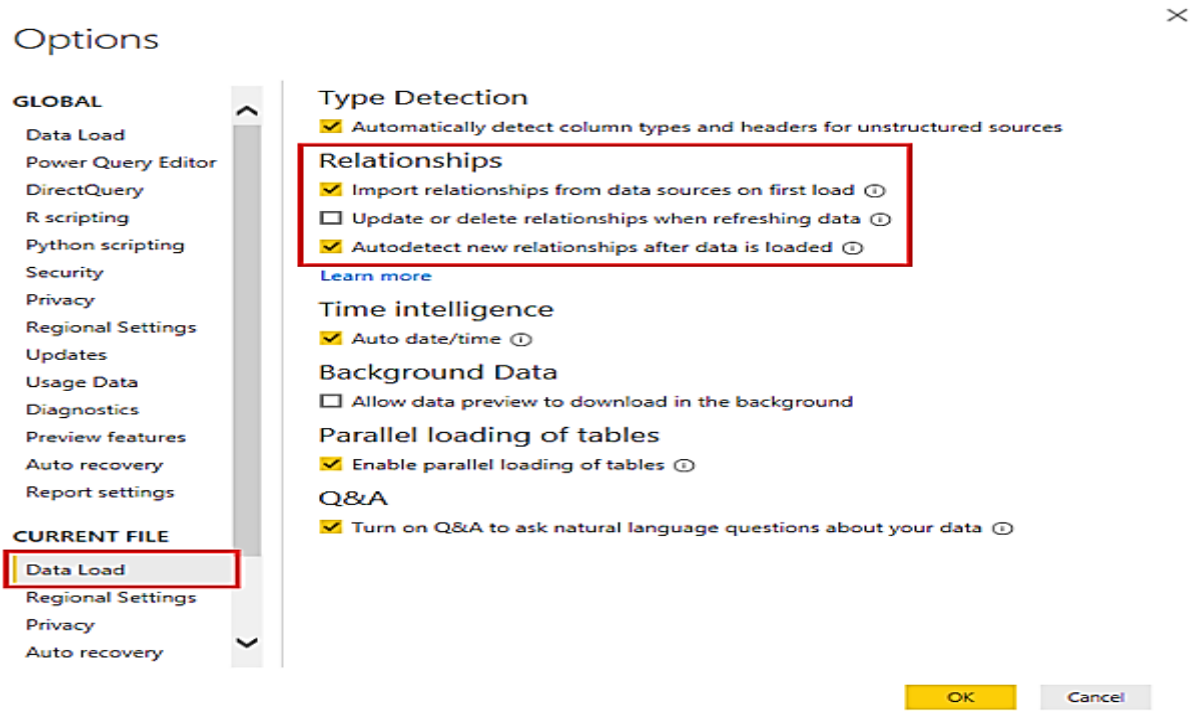
Cuando se crea una relación, ya sea con detección automática o de forma manual, Power BI Desktop configura automáticamente las opciones adicionales según los datos de las tablas. Estas propiedades de relación adicionales se encuentran en la parte inferior de los cuadros de diálogo Crear relación y Editar relación.



Power BI establece típicamente estas opciones automáticamente y no tendrá que ajustarlas. Si bien hay varias situaciones en las que podría querer configurar estas opciones por su cuenta.

Actualización de relaciones automáticas

Puede administrar cómo Power BI trata y ajusta automáticamente las relaciones en los informes y modelos. Para especificar cómo Power BI controla las opciones de relaciones, seleccione Archivo>Opciones y configuración>Opciones en Power BI Desktop y elija Carga de datos en el panel izquierdo. Aparecerán las opciones para Relaciones.



Hay tres opciones que se pueden seleccionar y habilitar:

- Importar relaciones de orígenes de datos en la primera carga: Esta opción está activada de forma predeterminada. Si se selecciona, Power BI comprobará las relaciones definidas en el origen de datos, tales como las relaciones entre la clave externa y la clave principal en el almacenamiento de datos. Si existen dichas relaciones, se reflejarán en el modelo de datos de Power BI al cargar los datos por primera vez. Esta opción permite empezar a trabajar rápidamente con el modelo, sin necesidad de tener que buscar o definir esas relaciones por su cuenta.

- Actualizar o eliminar las relaciones al actualizar los datos: Esta opción está desactivada de forma predeterminada. Si lo selecciona, Power BI comprueba los cambios en las relaciones de las fuentes de datos cuando se actualiza el modelo semántico. Si dichas relaciones cambian o se quitan, Power BI refleja esos cambios en su propio modelo de datos, y las actualiza o elimina para que coincidan.
- Detectar automáticamente nuevas relaciones después de cargar los datos: Esta opción se describe en Detección automática durante la carga.

Las futuras actualizaciones de datos requieren una cardinalidad diferente

Normalmente, Power BI Desktop puede determinar automáticamente la mejor cardinalidad para la relación. Si tiene que reemplazar la configuración automática, porque sabe que los datos cambiarán en el futuro, puede cambiarla en el control Cardinalidad. Veamos un ejemplo donde se debe seleccionar una cardinalidad diferente.

La tabla CompanyProjectPriority es una lista de todos los proyectos de la empresa y su prioridad. La tabla ProjectBudget es el conjunto de proyectos para los que se ha aprobado presupuesto.

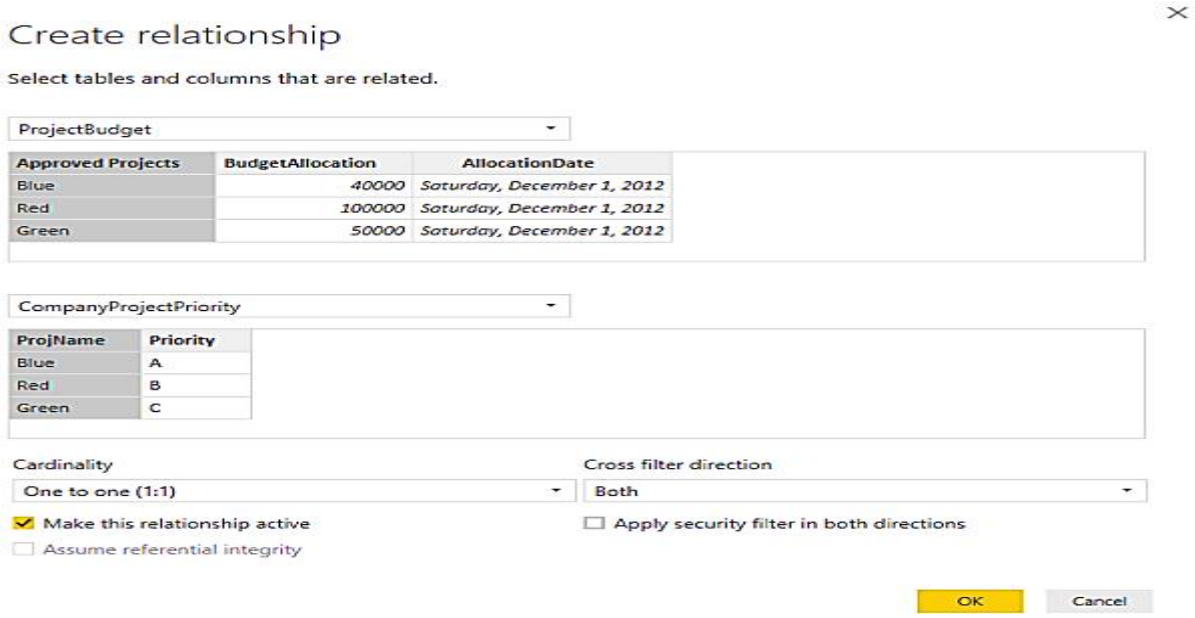
CompanyProjectPriority

ProjName	Priority
Azul	A
Rojo	B
Verde	C
Amarillo	C
Púrpura	B
Naranja	C

ProjectBudget

Proyectos aprobados	BudgetAllocation	AllocationDate
Azul	40,000	12/1/2012
Rojo	100,000	12/1/2012
Verde	50 000	12/1/2012

Si creamos una relación entre la columna ApprovedProjects de la tabla ProjectBudget y la columna ProjectName de la tabla CompanyProjectPriority, Power BI establece automáticamente Cardinalidad en Uno a uno (1:1) y Dirección de filtro cruzado en Ambos.



El motivo por el que Power BI crea esta configuración es porque, para Power BI Desktop, la mejor combinación de las dos tablas es la siguiente:

ProjName	Priority	BudgetAllocation	AllocationDate
Azul	A	40,000	12/1/2012
Rojo	B	100 000	12/1/2012
Verde	C	50,000	12/1/2012
Amarillo	C		
Púrpura	B		
Naranja	C		

Hay una relación uno a uno entre nuestras dos tablas porque no hay ningún valor que se repita en la columna ProjName de la tabla combinada. La columna ProjName es única porque cada valor se produce solo una vez, por lo que las filas de las dos tablas se pueden combinar directamente sin ninguna duplicación.

Sin embargo, supongamos que sabe que los datos cambiarán la próxima vez que los actualice. Una versión actualizada de la tabla ProjectBudget tiene ahora filas adicionales para los proyectos Azul y Rojo:

ProjectBudget

Proyectos aprobados	BudgetAllocation	AllocationDate
Azul	40,000	12/1/2012
Rojo	100,000	12/1/2012
Verde	50 000	12/1/2012
Azul	80,000	6/1/2013
Rojo	90,000	6/1/2013

Estas filas adicionales implican que la mejor combinación de las dos tablas tiene ahora el siguiente aspecto:

ProjName	Priority	BudgetAllocation	AllocationDate
Azul	A	40,000	12/1/2012
Rojo	B	100 000	12/1/2012
Verde	C	50,000	12/1/2012
Amarillo	C		
Púrpura	B		
Naranja	C		
Azul	A	80000	6/1/2013
Rojo	B	90000	6/1/2013

En esta nueva tabla combinada, la columna ProjName tiene valores repetidos. Las dos tablas originales no tienen una relación uno a uno, una vez que se actualiza la tabla. En este caso, como sabemos que las actualizaciones futuras harán que la columna ProjName tenga duplicados, queremos establecer Cardinalidad en Muchos a uno (*:1), donde el lado muchos se asignan a ProjectBudget y el lado uno, a CompanyProjectPriority.

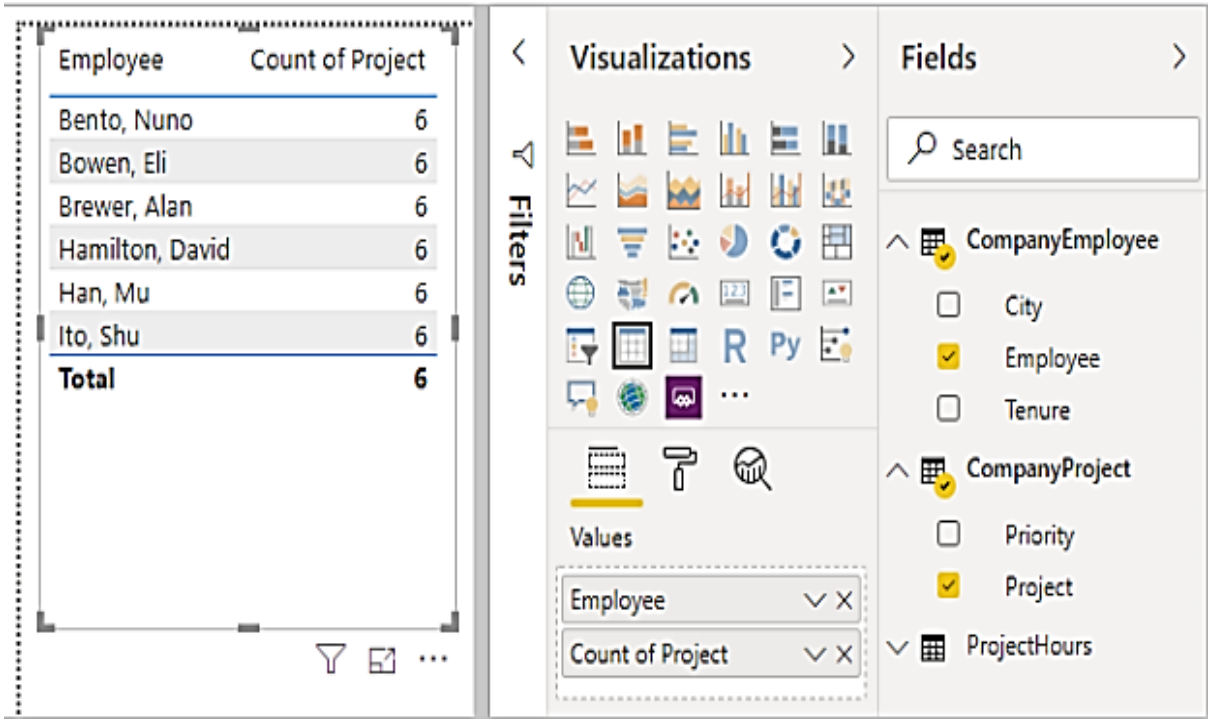
Ajuste de la dirección del filtro cruzado para un conjunto complejo de tablas y relaciones

Sin embargo, hay algunas circunstancias más poco frecuentes en las que es posible que tenga que establecer esta opción de forma diferente a la predeterminada. Un ejemplo es si importa un modelo de una versión más antigua de Power Pivot, donde cada relación se establece para una única dirección.

La opción Ambos permite que Power BI Desktop trate todos los aspectos de las tablas conectadas como si fueran una sola tabla. Sin embargo, existen algunas situaciones en las que Power BI Desktop no puede establecer la dirección de filtro cruzado de una relación en Ambos y también mantener un conjunto ambiguo de valores predeterminados disponibles a efectos de la elaboración de informes. Si la dirección de filtro cruzado de una relación no se establece en Ambos, suele ser porque se crearía ambigüedad. Si el valor predeterminado del filtro cruzado no funciona en su caso, intente configurarlo en una tabla determinada o en Ambos.

El filtro cruzado de una sola dirección funciona para muchas situaciones. De hecho, si ha importado un modelo de Power Pivot en Excel 2013 o versiones anteriores, todas las relaciones se establecerán en una dirección única. Una dirección única significa que las opciones de filtrado en tablas conectadas funcionan en la tabla donde sucede el trabajo de agregación. A veces, comprender el filtrado cruzado puede ser un poco complicado; veamos un ejemplo.

Con el filtro cruzado de dirección única, si crea un informe que resuma las horas del proyecto, podrá optar por resumir (o filtrar) por la tabla CompanyProject y su columna Priority o por la tabla CompanyEmployee y su columna City. Sin embargo, si quiere contar el número de empleados por proyecto (una pregunta menos común), esto no funcionará. Obtendrá una columna de valores que son los mismos. En el ejemplo siguiente, la dirección de filtro cruzado de ambas relaciones se establece en una dirección única: hacia la tabla ProjectHours. En el cuadro Valores, el campo Proyecto se establece en Recuento:



La especificación del filtro fluirá de CompanyProject a ProjectHours (como se muestra en la imagen siguiente), pero no hasta CompanyEmployee.



CompanyEmployee

Employ	Tenure	City
Brewer, A	15	Redmond
Bowen, El	10	San Jose
Bento, Nu	15	Redmond
Hamilton,	3	San Jose
Han, Mu	1	San Jose
Ito, Shu	1	Redmond

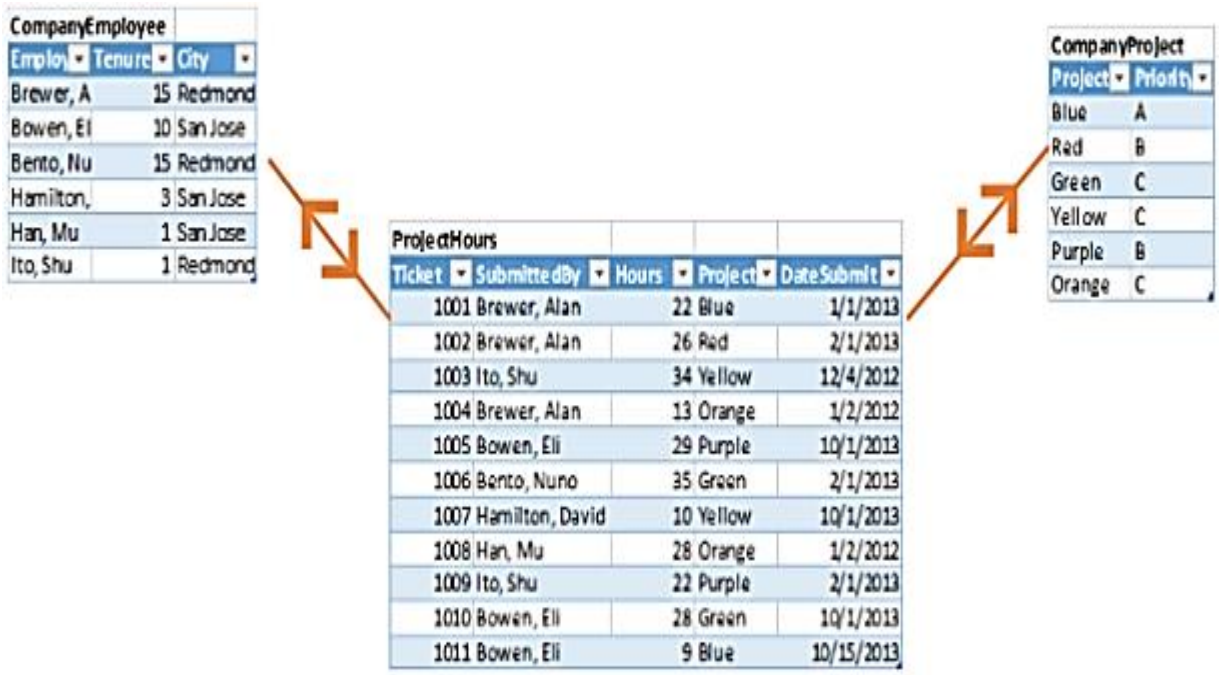
ProjectHours

Ticket	Submitted By	Hours	Project	Date Submit
1001	Brewer, Alan	22	Blue	1/1/2013
1002	Brewer, Alan	26	Red	2/1/2013
1003	Ito, Shu	34	Yellow	12/4/2012
1004	Brewer, Alan	13	Orange	1/2/2012
1005	Bowen, Eli	29	Purple	10/1/2013
1006	Bento, Nuno	35	Green	2/1/2013
1007	Hamilton, David	10	Yellow	10/1/2013
1008	Han, Mu	28	Orange	1/2/2012
1009	Ito, Shu	22	Purple	2/1/2013
1010	Bowen, Eli	28	Green	10/1/2013
1011	Bowen, Eli	9	Blue	10/15/2013

CompanyProject

Project	Priority
Blue	A
Red	B
Green	C
Yellow	C
Purple	B
Orange	C

Sin embargo, si establece la dirección de filtro cruzado en Ambos, sí funcionará. La opción Ambos permite que la especificación del filtro fluya hasta CompanyEmployee.



CompanyEmployee

Employ	Tenure	City
Brewer, A	15	Redmond
Bowen, El	10	San Jose
Bento, Nu	15	Redmond
Hamilton,	3	San Jose
Han, Mu	1	San Jose
Ito, Shu	1	Redmond

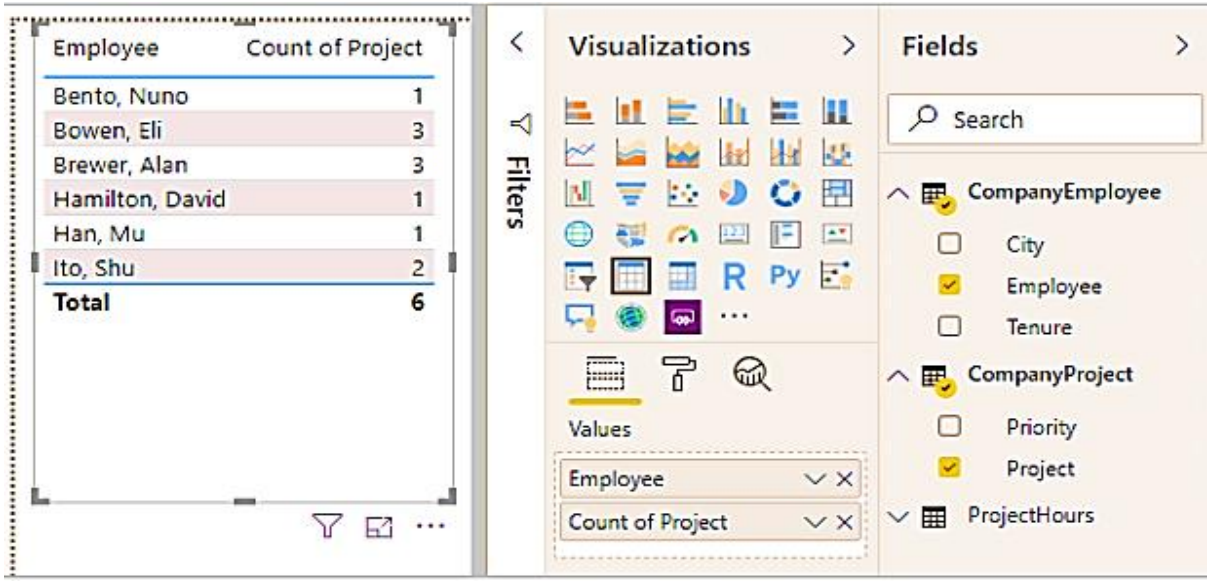
ProjectHours

Ticket	SubmittedBy	Hours	Project	DateSubmit
1001	Brewer, Alan	22	Blue	1/1/2013
1002	Brewer, Alan	26	Red	2/1/2013
1003	Ito, Shu	34	Yellow	12/4/2012
1004	Brewer, Alan	13	Orange	1/2/2012
1005	Bowen, Eli	29	Purple	10/1/2013
1006	Bento, Nuno	35	Green	2/1/2013
1007	Hamilton, David	10	Yellow	10/1/2013
1008	Han, Mu	28	Orange	1/2/2012
1009	Ito, Shu	22	Purple	2/1/2013
1010	Bowen, Eli	28	Green	10/1/2013
1011	Bowen, Eli	9	Blue	10/15/2013

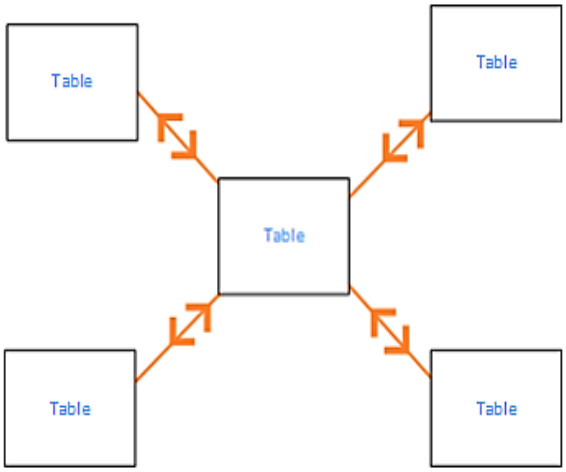
CompanyProject

Project	Priority
Blue	A
Red	B
Green	C
Yellow	C
Purple	B
Orange	C

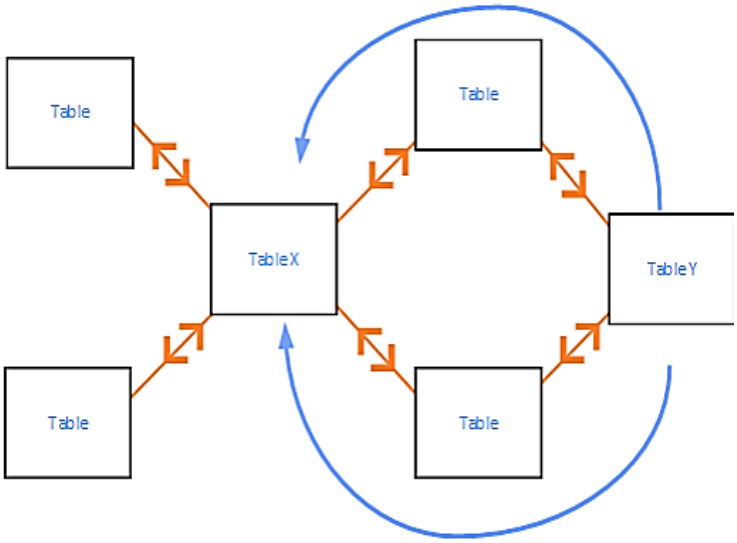
Con la dirección de filtro cruzado establecida en Ambos, nuestro informe parece ahora correcto:



El filtrado cruzado en ambas direcciones funciona bien con un modelo de relaciones de tablas como el patrón mostrado anteriormente. Este esquema se denomina habitualmente esquema de estrella, similar al siguiente:



La dirección del filtro cruzado no funciona bien con un patrón más general que se suele encontrarse en las bases de datos, como en este diagrama:



Si tiene un patrón de tabla como este, con bucles, el filtro cruzado puede crear un conjunto ambiguo de relaciones. Por ejemplo, si suma un campo de Tabla X y después elige filtrar por un campo en la Tabla Y, no resulta claro cómo debe viajar el filtro, si a través de la tabla superior o la tabla inferior. Un ejemplo común de este tipo de patrón se presenta con TableX como tabla de ventas con datos reales y TableY como tabla de datos de presupuesto. A continuación, las tablas en la parte central son tablas de búsqueda que utilizan las dos tablas, como la de división o de región.

De igual modo que con las relaciones activas e inactivas, Power BI Desktop no permitirá que una relación se establezca en Ambos si eso crease ambigüedad en los informes.

Hay varias maneras de controlar esta situación. Estas son las dos más comunes:

Eliminar o marcar las relaciones como inactivas para reducir la ambigüedad. A continuación, puede establecer un filtro cruzado de relación en Ambos.

Agregar una tabla dos veces (con un nombre diferente la segunda vez) para eliminar los bucles. Esto hace que el patrón de relaciones sea como un esquema de estrella. Con un esquema de estrella, todas las relaciones pueden establecerse en Ambos.

Relación activa incorrecta

Cuando Power BI Desktop crea automáticamente relaciones, a veces encuentra más de una relación entre dos tablas. Cuando esto sucede, solo una de las relaciones se establece para estar activa. La relación activa actúa como la relación predeterminada para que, al elegir los campos de dos tablas diferentes, Power BI Desktop pueda crear automáticamente una visualización por usted. Sin embargo, en algunos casos la relación seleccionada de forma automática puede ser incorrecta. Puede usar el cuadro de diálogo Administrar relaciones para establecer una relación como activa o inactiva, o bien establecer la relación activa en el cuadro de diálogo Editar relación.

Para garantizar que haya una relación predeterminada, Power BI Desktop solo permite una sola relación activa entre dos tablas en un momento dado. Por lo tanto, debe establecer primero la relación actual como inactiva y, luego, establecer la relación que quiere que esté activa.

Veamos un ejemplo. La primera tabla es ProjectTickets y la segunda tabla es EmployeeRole.

ProjectTickets

Ticket	OpenedBy	SubmittedBy	Hours	Project	DateSubmit
1001	Perham, Tom	Brewer, Alan	22	Azul	1/1/2013
1002	Roman, Daniel	Brewer, Alan	26	Rojo	2/1/2013
1003	Roth, Daniel	Ito, Shu	34	Amarillo	12/4/2012
1004	Perham, Tom	Brewer, Alan	13	Naranja	1/2/2012
1005	Roman, Daniel	Bowen, Eli	29	Púrpura	1/10/2013
1006	Roth, Daniel	Bento, Nuno	35	Verde	2/1/2013
1007	Roth, Daniel	Hamilton, David	10	Amarillo	1/10/2013
1008	Perham, Tom	Han, Mu	28	Naranja	1/2/2012
1009	Roman, Daniel	Ito, Shu	22	Púrpura	2/1/2013
1010	Roth, Daniel	Bowen, Eli	28	Verde	10/1/2013
1011	Perham, Tom	Bowen, Eli	9	Azul	10/15/2013

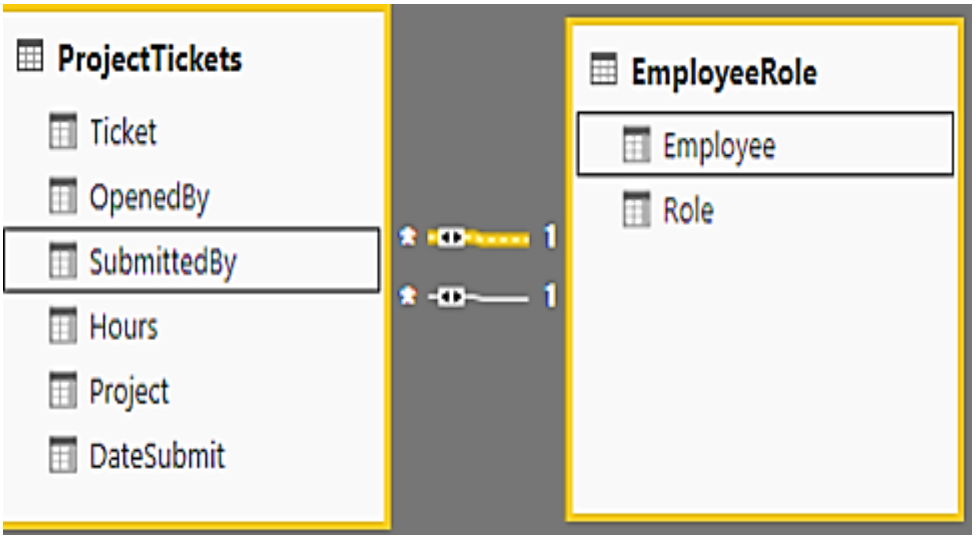
EmployeeRole

Employee	Role
Bento, Nuno	Administrador del proyecto
Bowen, Eli	Responsable del proyecto
Brewer, Alan	Administrador del proyecto
Hamilton, David	Responsable del proyecto
Han, Mu	Responsable del proyecto
Ito, Shu	Responsable del proyecto
Perham, Tom	Patrocinador del proyecto
Roman, Daniel	Patrocinador del proyecto
Roth, Daniel	Patrocinador del proyecto

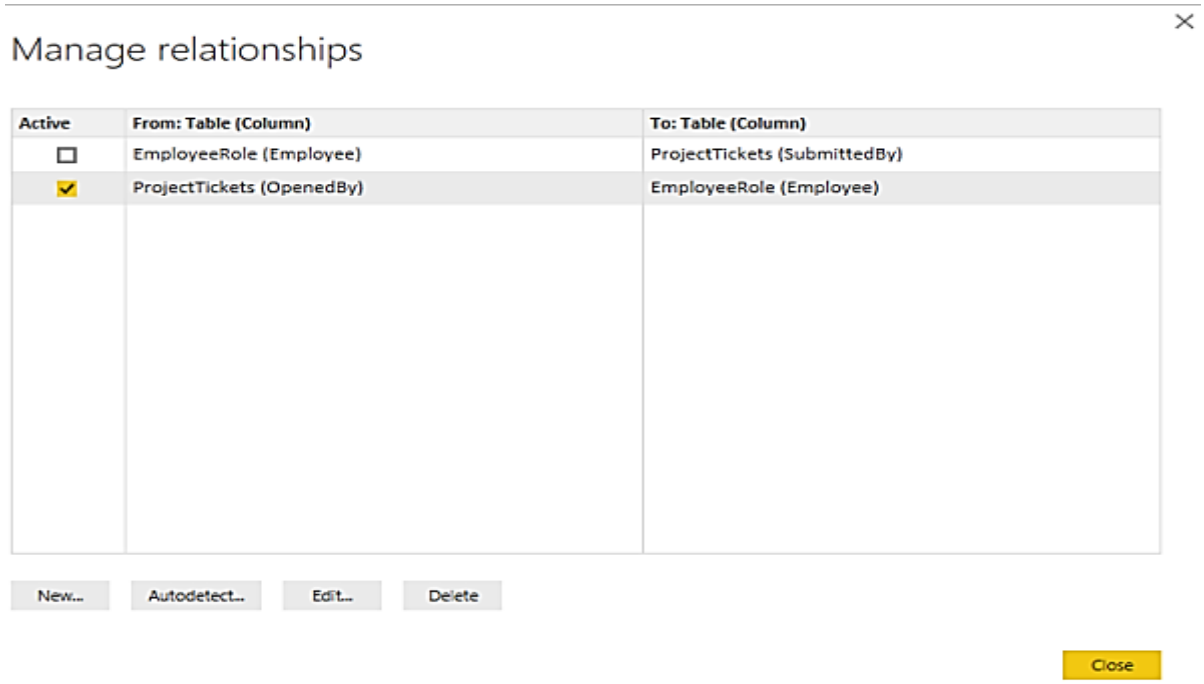
En realidad, hay dos relaciones aquí:

Entre Employee en la tabla EmployeeRole y SubmittedBy en la tabla ProjectTickets.

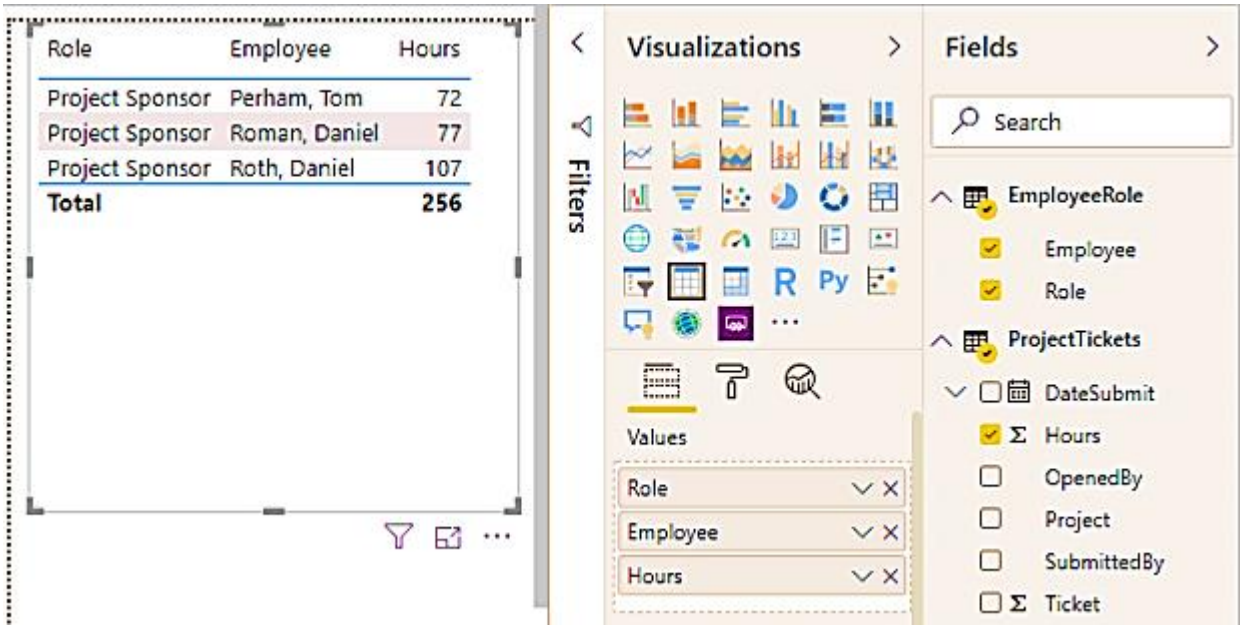
Entre OpenedBy en la tabla ProjectTickets y Employee en la tabla EmployeeRole.



Si agregamos ambas relaciones al modelo (OpenedBy primero), el cuadro de diálogo Administrar relaciones mostrará que OpenedBy está activa:



Ahora, si creamos un informe que usa los campos Role y Employee de EmployeeRole y el campo Hours de ProjectTickets en una visualización de la tabla en el lienzo del informe, solo veremos patrocinadores del proyecto porque son los únicos que abrieron un vale del proyecto.



Podemos cambiar la relación activa y obtener SubmittedBy en lugar de OpenedBy. En Administrar relaciones, desactivamos la relación ProjectTickets(OpenedBy) con EmployeeRole(Employee) y, luego, activamos la relación EmployeeRole(Employee) con Project Tickets(SubmittedBy).

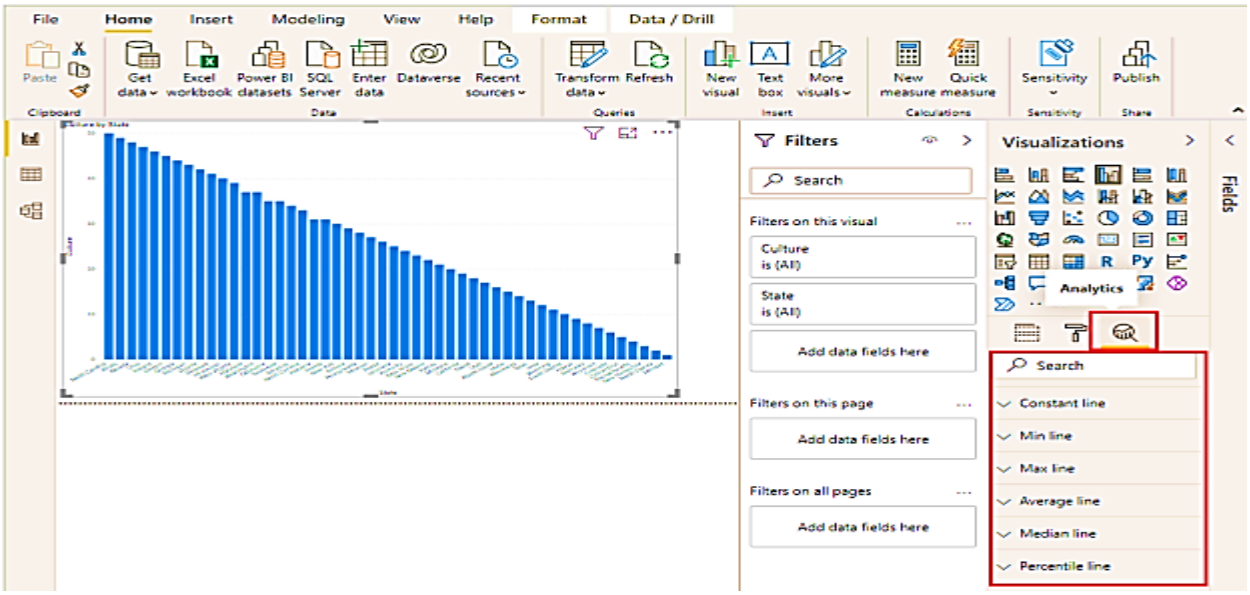


Consulta de todas las relaciones en la vista Relación

A veces, el modelo tiene varias tablas y relaciones complejas entre ellas. La vista Relación de Power BI Desktop muestra todas las relaciones del modelo, su dirección y cardinalidad en un diagrama personalizable y fácil de entender.

Uso del panel Análisis en Power BI Desktop

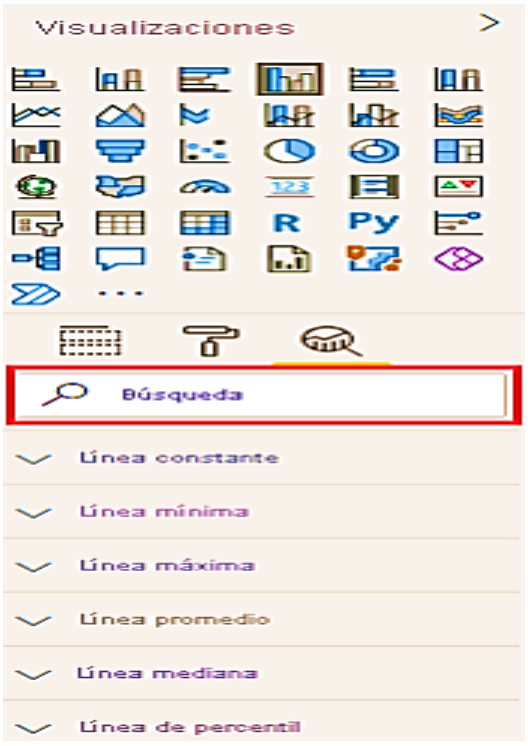
Con el panel Análisis en Power BI Desktop, puede agregar líneas de referencia dinámicas en objetos visuales y destacar las tendencias o detalles importantes. El panel e icono Análisis se encuentran en el área Visualizaciones de Power BI Desktop.



Nota: El panel Analytics solo se muestra cuando se selecciona un objeto visual en el lienzo de Power BI Desktop.

Búsqueda en el panel Análisis

Puede buscar en el panel Análisis, que es una subsección del panel Visualizaciones. El cuadro de búsqueda aparece cuando se selecciona el icono Análisis.



Uso del panel Analytics

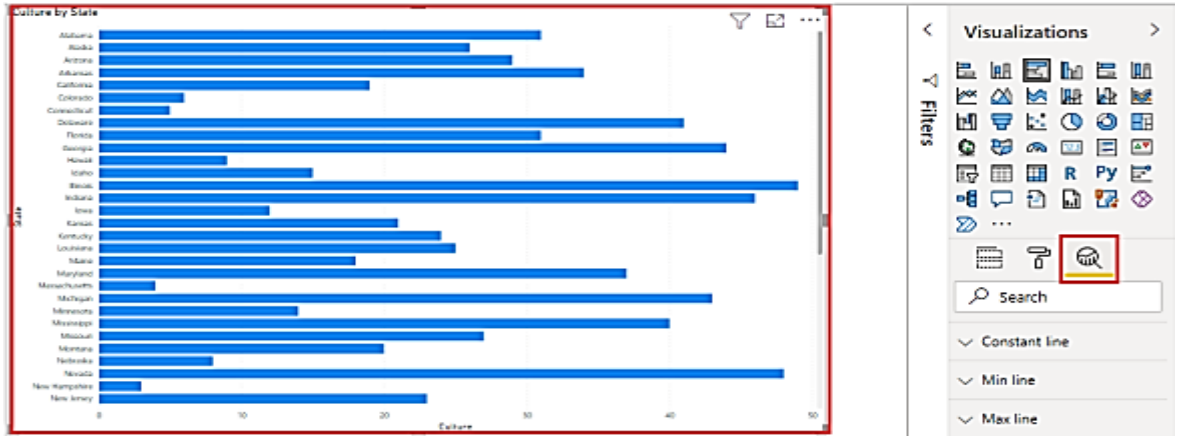
Con el panel Análisis, puede crear estos tipos de líneas de referencias dinámicas:

- Línea constante del eje X
- Línea constante del eje Y
- Línea mínima
- Línea máxima
- Línea promedio
- Línea mediana
- Línea de percentil
- Sombreado de simetría

Las secciones siguientes muestran cómo se puede usar el panel Analytics y las líneas de referencia dinámicas en las visualizaciones.

Para ver las líneas de referencia dinámicas disponibles para un objeto visual, siga estos pasos:

1. Seleccione o cree un objeto visual y, a continuación, seleccione el icono Analytics desde la sección Visualizaciones.

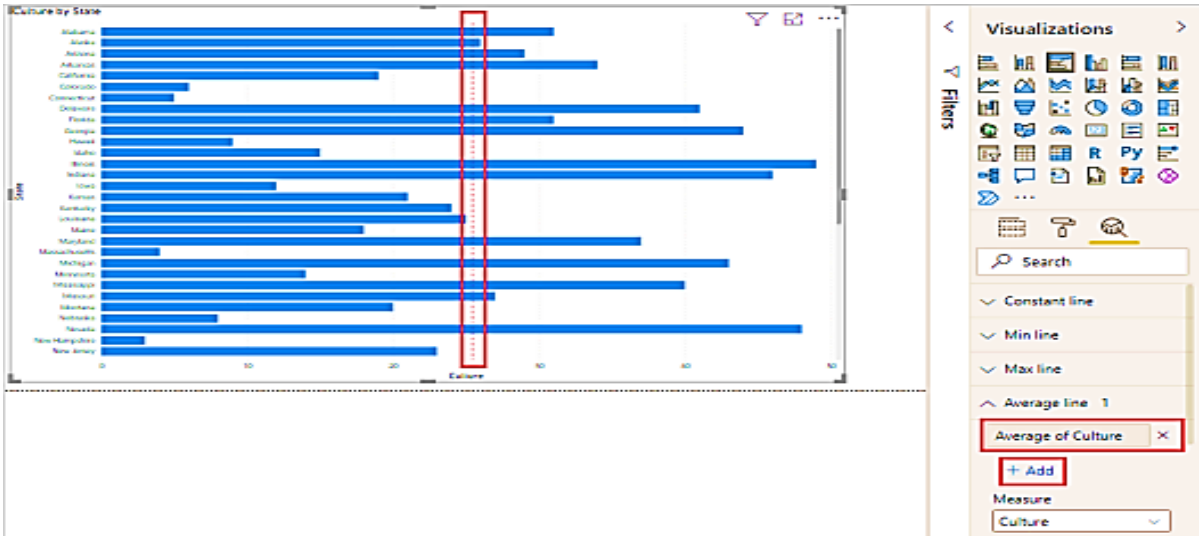


2. Seleccione el tipo de línea que desea crear para expandir sus opciones. En este ejemplo se muestra el elemento Línea de promedio seleccionado.

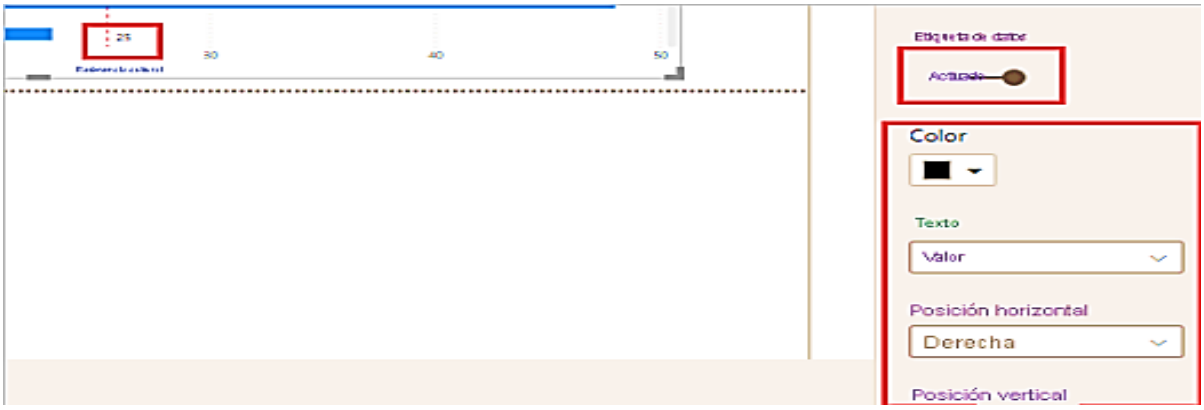


3. Para crear una nueva línea, seleccione + Agregar. Luego, puede asignarle un nombre a la línea. Haga doble clic en el cuadro de texto y escriba su nombre.

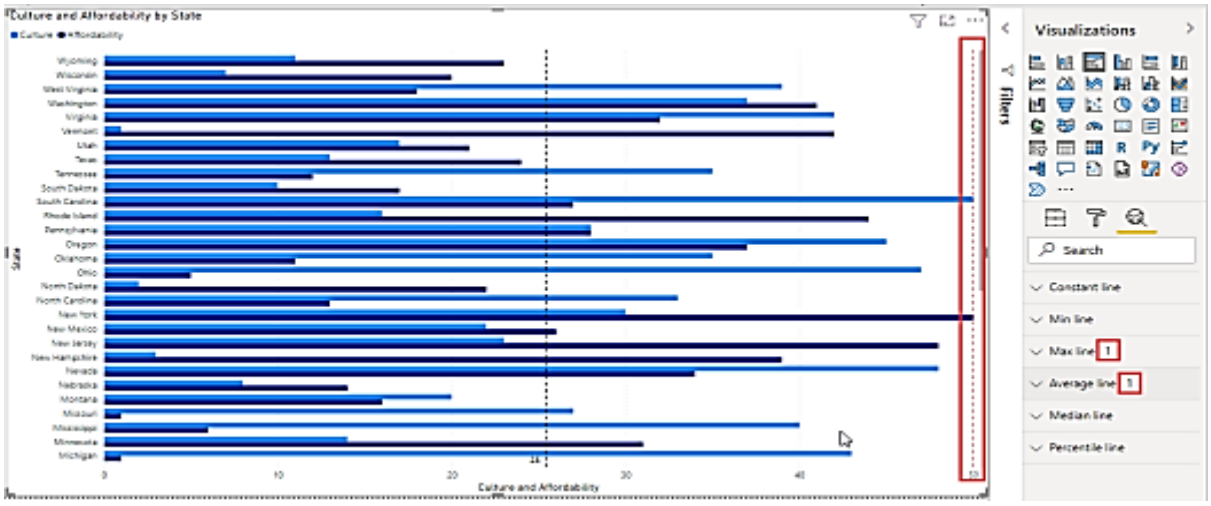
Ahora tiene todo tipo de opciones para la línea. Puede especificar su Color, el porcentaje de Transparencia, el Estilo de línea y la Posición (en comparación con los elementos de datos del objeto visual). También puede elegir si quiere incluir la Etiqueta de datos. Para especificar la medida visual en la que se basará la línea, seleccione la lista desplegable Medida, que se rellena automáticamente con los elementos de datos del objeto visual. En este ejemplo se selecciona Referencia cultural como medida y se etiqueta Average of Culture (Promedio de cultura). Verá cómo personalizar algunas de las otras opciones en los pasos posteriores.



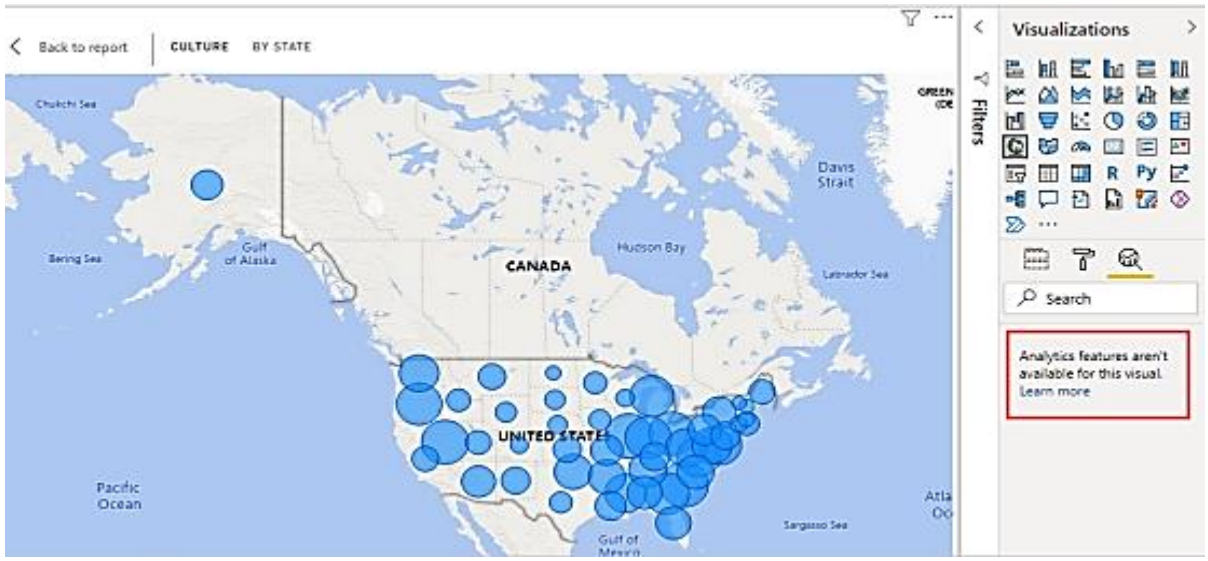
4. Si quiere que aparezca una etiqueta de datos, cambie Etiqueta de datos de Desactivada a Activada. Cuando lo haga, obtendrá muchas más opciones para la etiqueta de datos.



5. Observe el número que aparece junto al elemento Línea promedio en el panel Analytics. Eso le indica cuántas líneas dinámicas tiene actualmente en el objeto visual y de qué tipo son. Si agregamos una Línea máxima para Affordability (Asequibilidad), el panel Análisis muestra que ahora también tenemos una línea de referencia dinámica de Línea máxima aplicada a este objeto visual.



Si el objeto visual seleccionado no puede tener líneas de referencia dinámica aplicadas (en este caso, un objeto visual de Mapa), verá el mensaje siguiente al seleccionar el panel Análisis.



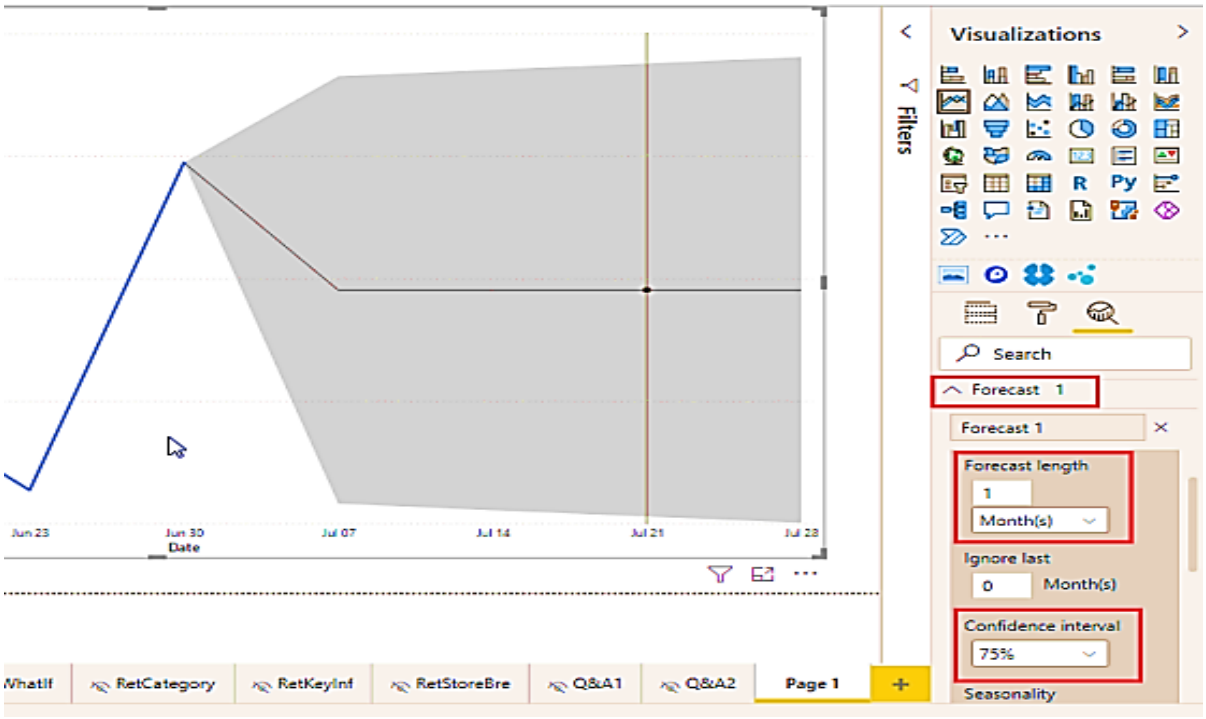
Puede resaltar distintos tipos de información interesante mediante la creación de líneas de referencia dinámicas con el panel Análisis.

Se están planeando más características y funcionalidades, como la ampliación de los objetos visuales que pueden tener líneas de referencia dinámicas aplicadas. Vuelva periódicamente para ver las novedades.

Aplicación de Previsión

Si tiene datos de tiempo en el origen, puede usar la característica de previsión. Seleccione un objeto visual y, luego, expanda la sección Previsión del panel Análisis.

Puede especificar muchas entradas para modificar la previsión, como Predecir duración o el Intervalo de confianza. En la imagen siguiente se muestra un objeto visual de línea básica con la previsión aplicada. Use su imaginación (y juegue con la previsión) para ver cómo se puede aplicar a los modelos.



Trabajo con Vista de datos en Power BI Desktop

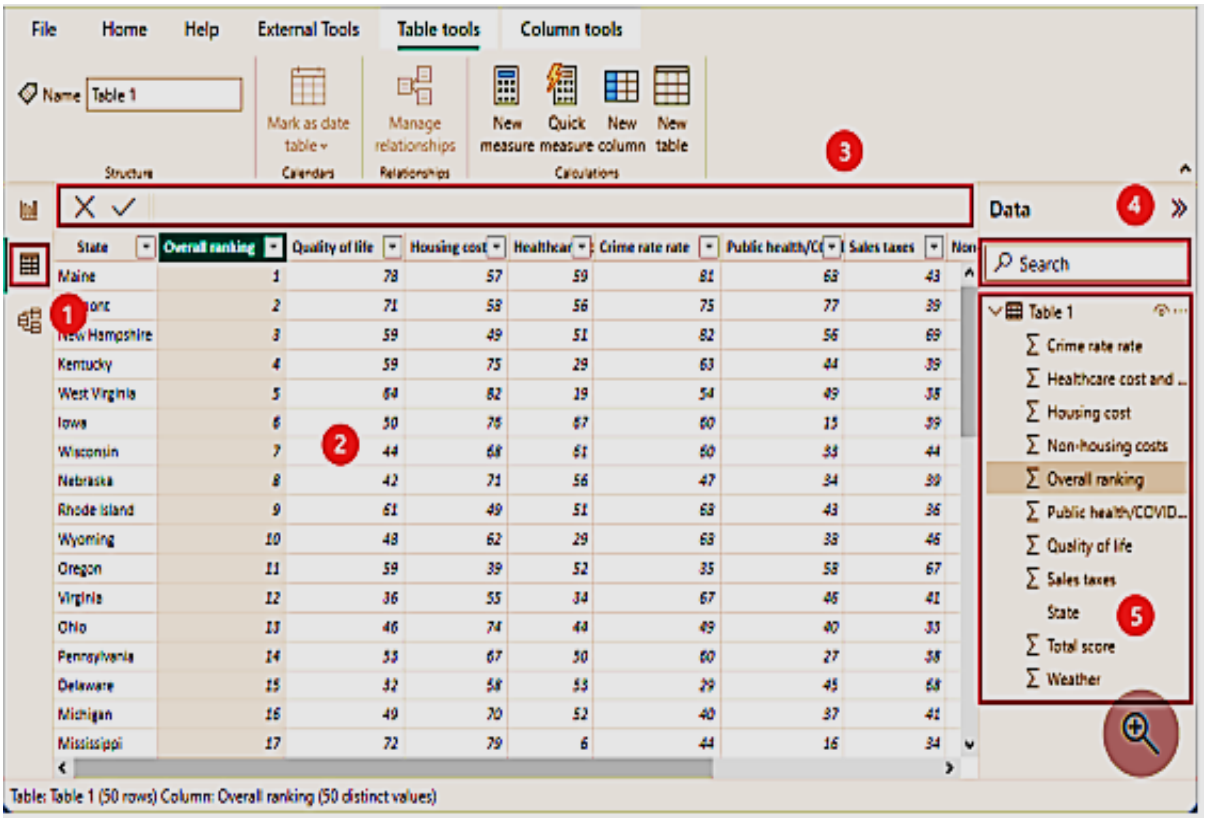
La vista Datos permite inspeccionar, explorar y analizar los datos de su modelo de Power BI Desktop. Es distinto de cómo se visualizan las tablas, las columnas y los datos en el Editor de Power Query. Con la vista Datos, se ven los datos después de que se hayan cargado en el modelo.

Nota: Dado que la vista Datos muestra los datos después de haberlos cargado en el modelo, el icono de la vista Datos no está visible si todos los orígenes de datos se basan en DirectQuery.

Cuando se están modelando los datos, a veces quiere ver lo que hay realmente en una tabla o columna sin crear un objeto visual en el lienzo del informe. Es posible que quiera

ver hasta el nivel de fila. Esta capacidad es útil sobre todo cuando crea columnas calculadas y medidas o cuando tiene que identificar un tipo de datos o una categoría de datos.

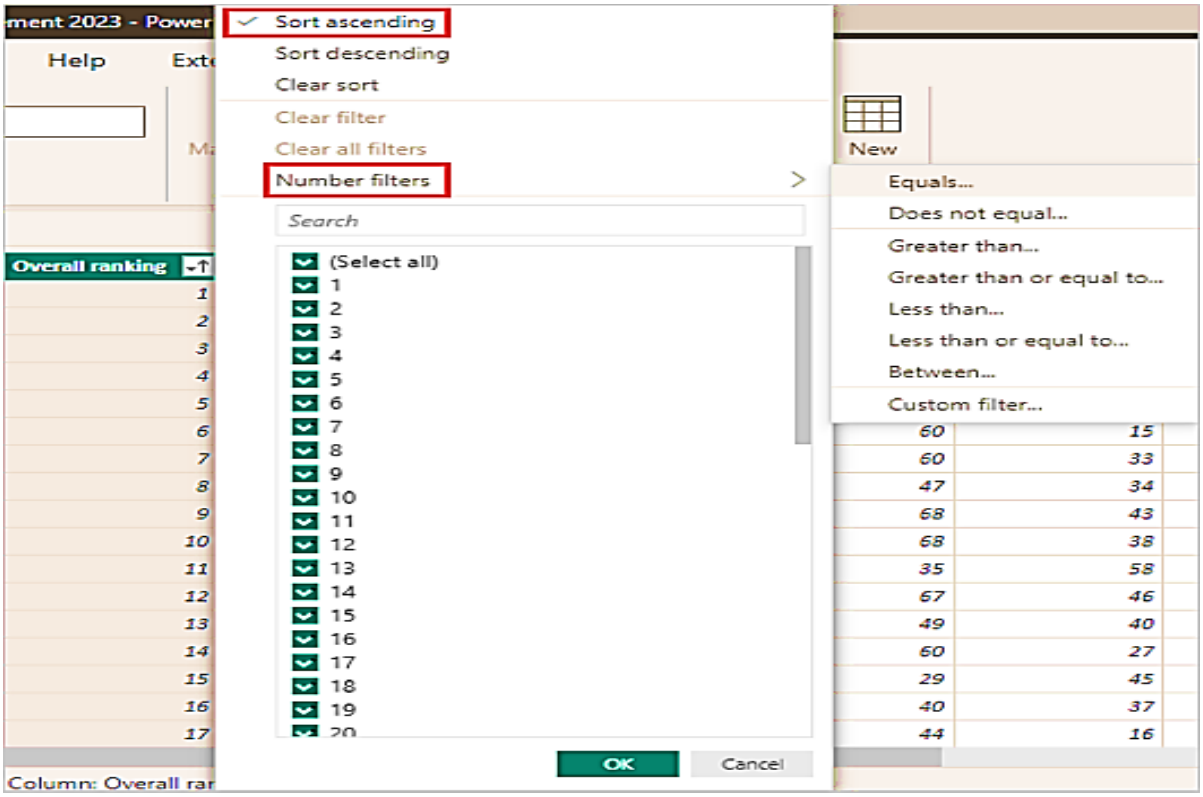
Eche un vistazo más de cerca a algunos de los elementos de la vista Datos.



1. Icono de la vista Datos. Seleccione este icono para entrar en la vista Datos.
2. Cuadrícula de datos. Esta área muestra la tabla seleccionada y todas las columnas y filas que hay en ella. Las columnas ocultas en la vista Informe están atenuadas. Puede hacer clic con el botón derecho en una columna para ver las opciones.
3. Barra de fórmulas. Introduzca fórmulas de expresiones de análisis de datos (DAX) para las columnas calculadas y las medidas.
4. Búsqueda. Busque una tabla o una columna en el modelo.
5. Lista de campos. Seleccione una tabla o una columna para verla en la cuadrícula de datos.

Filtrado en la vista Datos

También se pueden filtrar y ordenar datos en la vista Datos. En cada columna se muestra un icono que identifica la dirección de ordenación, si procede.

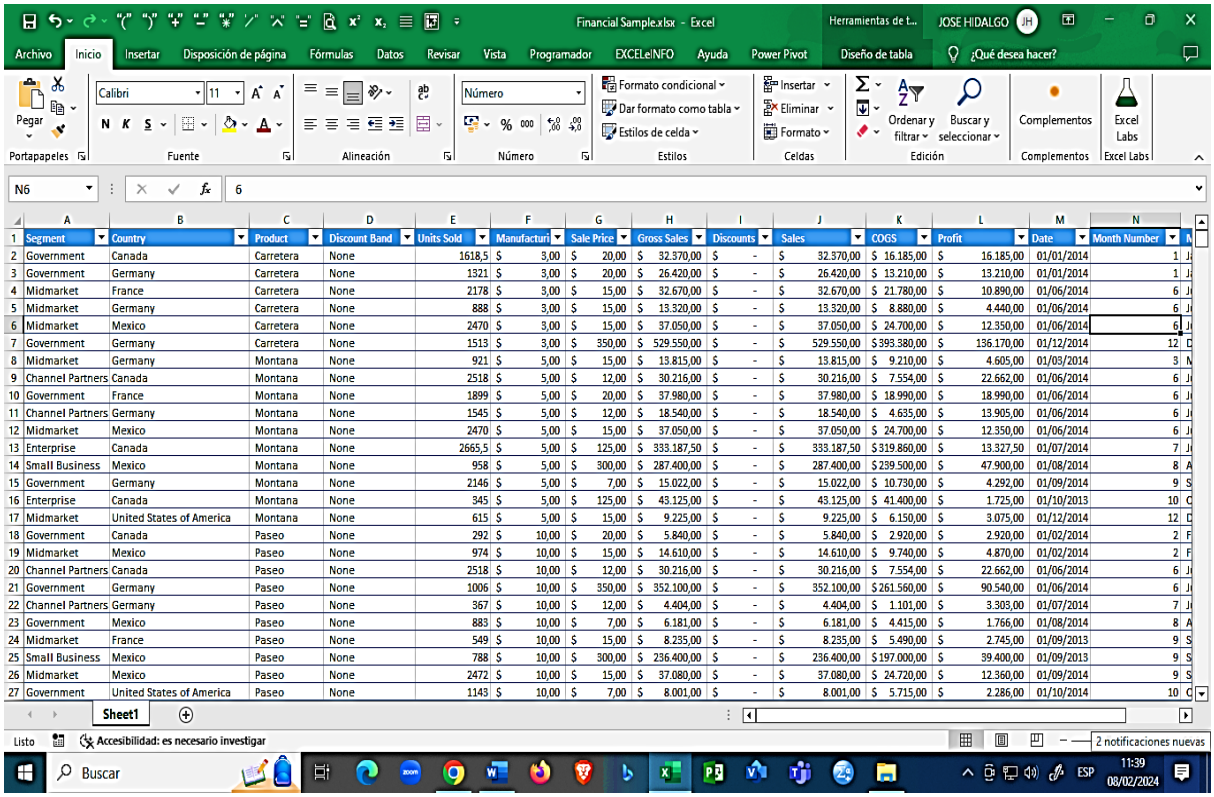


Puede filtrar valores concretos o puede usar el filtrado avanzado en función de los datos de la columna.

Ejemplo de aplicación en Power BI

Para este ejemplo de aplicación, se utiliza un archivo en Excel, denominado: Financial Sample.xlsx

El mismo que tiene una hoja con información financiera; a continuación, un extracto gráfico de la información que contiene el mencionado archivo:



Segment	Country	Product	Discount Band	Units Sold	Manufacturi	Sale Price	Gross Sales	Discounts	Sales	COGS	Profit	Date	Month Number
Government	Canada	Carretera	None	1618,5	$ 3,00	$ 20,00	$ 32.370,00	$ -	$ 32.370,00	$ 16.185,00	$ 16.185,00	01/01/2014	1
Government	Germany	Carretera	None	1321	$ 3,00	$ 20,00	$ 26.420,00	$ -	$ 26.420,00	$ 13.210,00	$ 13.210,00	01/01/2014	1
Midmarket	France	Carretera	None	2178	$ 3,00	$ 15,00	$ 32.670,00	$ -	$ 32.670,00	$ 21.780,00	$ 10.890,00	01/06/2014	6
Midmarket	Germany	Carretera	None	888	$ 3,00	$ 15,00	$ 13.320,00	$ -	$ 13.320,00	$ 8.880,00	$ 4.440,00	01/06/2014	6
Midmarket	Mexico	Carretera	None	2470	$ 3,00	$ 15,00	$ 37.050,00	$ -	$ 37.050,00	$ 24.700,00	$ 12.350,00	01/06/2014	6
Government	Germany	Carretera	None	1513	$ 3,00	$ 350,00	$ 529.550,00	$ -	$ 529.550,00	$ 393.380,00	$ 136.170,00	01/12/2014	12
Midmarket	Germany	Montana	None	921	$ 5,00	$ 15,00	$ 13.815,00	$ -	$ 13.815,00	$ 9.210,00	$ 4.605,00	01/03/2014	3
Channel Partners	Canada	Montana	None	2518	$ 5,00	$ 12,00	$ 30.216,00	$ -	$ 30.216,00	$ 7.554,00	$ 22.662,00	01/06/2014	6
Government	France	Montana	None	1899	$ 5,00	$ 20,00	$ 37.980,00	$ -	$ 37.980,00	$ 18.990,00	$ 18.990,00	01/06/2014	6
Channel Partners	Germany	Montana	None	1545	$ 5,00	$ 12,00	$ 18.540,00	$ -	$ 18.540,00	$ 4.635,00	$ 13.905,00	01/06/2014	6
Midmarket	Mexico	Montana	None	2470	$ 5,00	$ 15,00	$ 37.050,00	$ -	$ 37.050,00	$ 24.700,00	$ 12.350,00	01/06/2014	6
Enterprise	Canada	Montana	None	2665,5	$ 5,00	$ 125,00	$ 333.187,50	$ -	$ 333.187,50	$ 319.860,00	$ 13.327,50	01/07/2014	7
Small Business	Mexico	Montana	None	958	$ 5,00	$ 300,00	$ 287.400,00	$ -	$ 287.400,00	$ 239.500,00	$ 47.900,00	01/08/2014	8
Government	Germany	Montana	None	2146	$ 5,00	$ 7,00	$ 15.022,00	$ -	$ 15.022,00	$ 10.730,00	$ 4.292,00	01/09/2014	9
Enterprise	Canada	Montana	None	345	$ 5,00	$ 125,00	$ 43.125,00	$ -	$ 43.125,00	$ 41.400,00	$ 1.725,00	01/10/2013	10
Midmarket	United States of America	Montana	None	615	$ 5,00	$ 15,00	$ 9.225,00	$ -	$ 9.225,00	$ 6.150,00	$ 3.075,00	01/12/2014	12
Government	Canada	Paseo	None	292	$ 10,00	$ 20,00	$ 5.840,00	$ -	$ 5.840,00	$ 2.920,00	$ 2.920,00	01/02/2014	2
Midmarket	Mexico	Paseo	None	974	$ 10,00	$ 15,00	$ 14.610,00	$ -	$ 14.610,00	$ 9.740,00	$ 4.870,00	01/02/2014	2
Channel Partners	Canada	Paseo	None	2518	$ 10,00	$ 12,00	$ 30.216,00	$ -	$ 30.216,00	$ 7.554,00	$ 22.662,00	01/06/2014	6
Government	Germany	Paseo	None	1006	$ 10,00	$ 350,00	$ 352.100,00	$ -	$ 352.100,00	$ 261.560,00	$ 90.540,00	01/06/2014	6
Channel Partners	Germany	Paseo	None	367	$ 10,00	$ 12,00	$ 4.404,00	$ -	$ 4.404,00	$ 1.101,00	$ 3.303,00	01/07/2014	7
Government	Mexico	Paseo	None	883	$ 10,00	$ 7,00	$ 6.181,00	$ -	$ 6.181,00	$ 4.415,00	$ 1.766,00	01/08/2014	8
Midmarket	France	Paseo	None	549	$ 10,00	$ 15,00	$ 8.235,00	$ -	$ 8.235,00	$ 5.490,00	$ 2.745,00	01/09/2013	9
Small Business	Mexico	Paseo	None	788	$ 10,00	$ 300,00	$ 236.400,00	$ -	$ 236.400,00	$ 197.000,00	$ 39.400,00	01/09/2013	9
Midmarket	Mexico	Paseo	None	2472	$ 10,00	$ 15,00	$ 37.080,00	$ -	$ 37.080,00	$ 24.720,00	$ 12.360,00	01/09/2014	9
Government	United States of America	Paseo	None	1143	$ 10,00	$ 7,00	$ 8.001,00	$ -	$ 8.001,00	$ 5.715,00	$ 2.286,00	01/10/2014	10

Esta información para ser procesada, se requiere ser subida utilizando Power BI.

La jefa quiere ver un informe sobre las cifras de ventas más recientes. Ha solicitado un resumen ejecutivo de lo siguiente:

- Mes y año en los que se ha conseguido el mayor beneficio
- Ubicaciones donde la empresa tiene más éxito (por país/región)
- Producto y segmento en los que la empresa debe seguir centrando su inversión

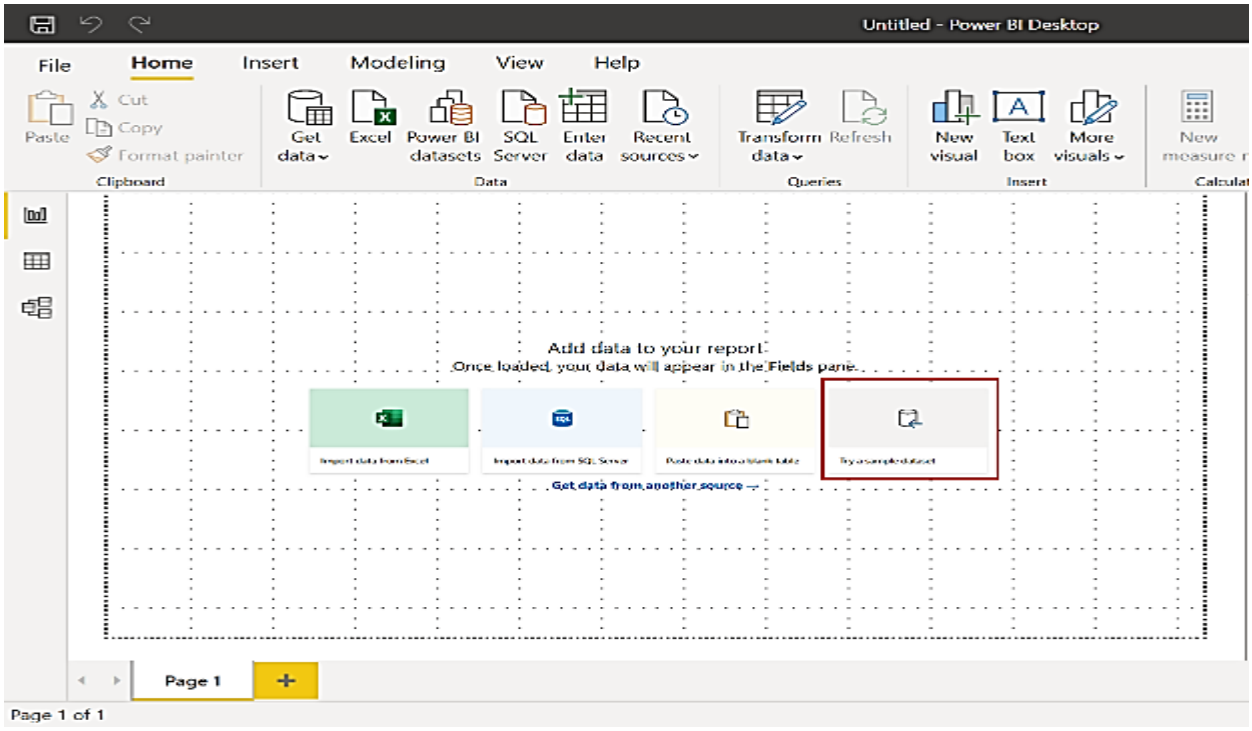
Con el libro de ejemplos financieros, este informe se puede crear sin dificultad. El informe final tendrá este aspecto.

Para realizar lo solicitado, hay que:

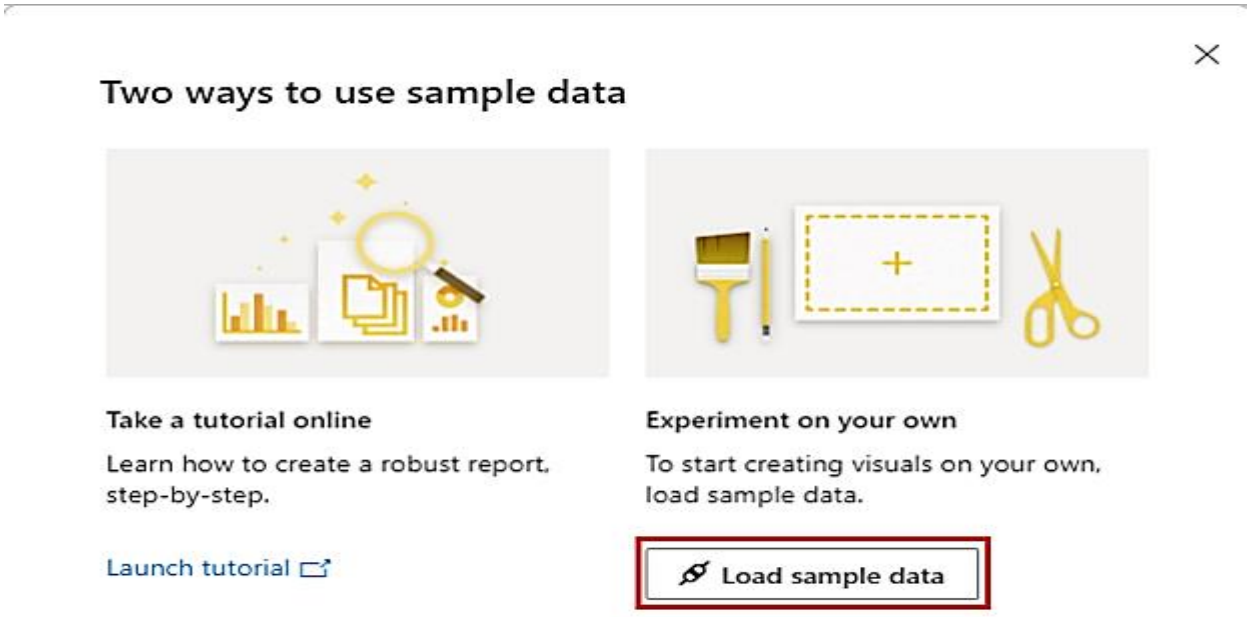
- Preparar los datos con algunas transformaciones.
- Crear un informe con un título, tres objetos visuales y una segmentación.
- Publicar el informe en el servicio Power BI para compartirlo con los compañeros de trabajo.

Obtención de datos en Power BI Desktop

Cuando abra Power BI Desktop, seleccione Probar un modelo semántico de ejemplo en el lienzo en blanco.



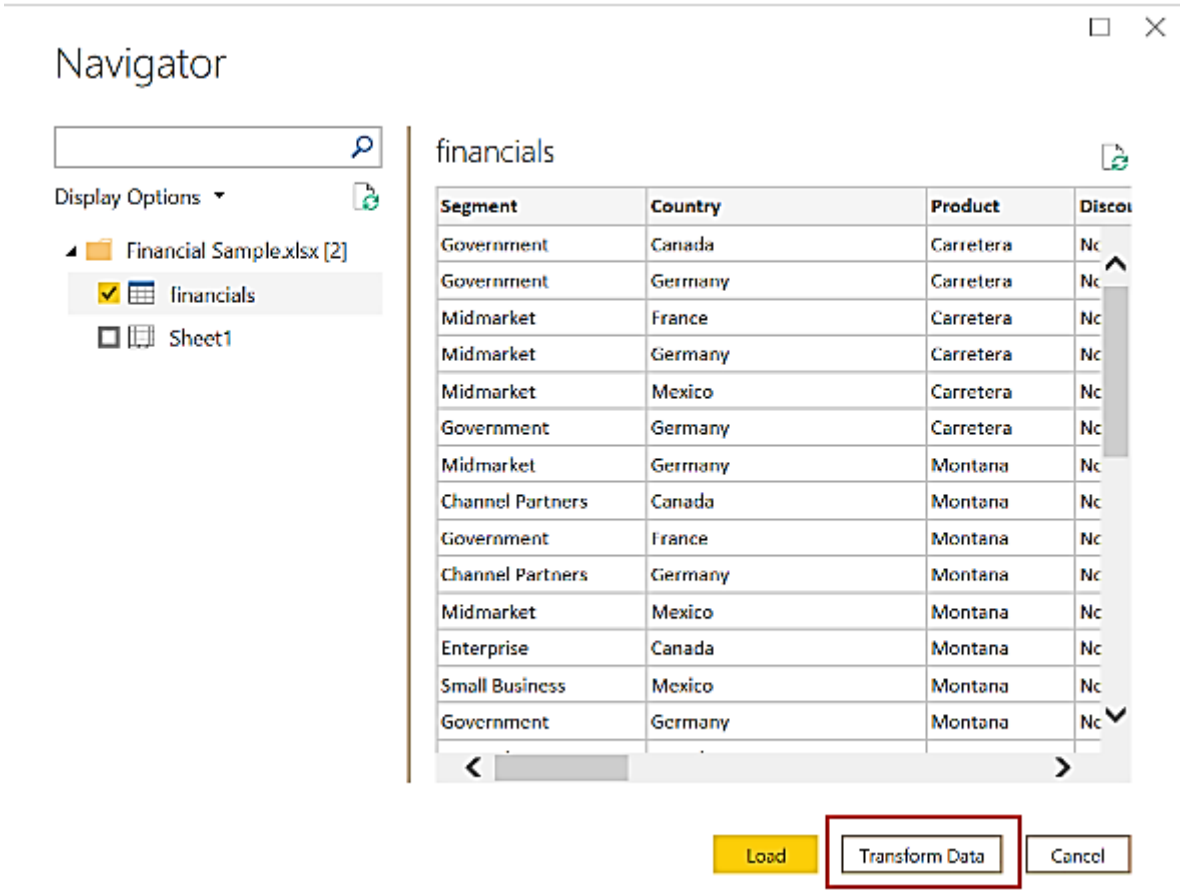
continúe y elija Cargar datos.



Preparar los datos

En Navegador, tiene la opción de transformar o cargar los datos. El Navegador proporciona una vista previa de los datos para que pueda comprobar que tiene el intervalo correcto. Los tipos de datos numéricos se muestran en cursiva. Si tiene que realizar cambios, transforme los datos antes de cargarlos. Para que las visualizaciones sean más fáciles de leer más adelante, le interesa transformar los datos ahora. A medida que realiza cada transformación, verá que se agrega a la lista en Configuración de la consulta, en Pasos aplicados

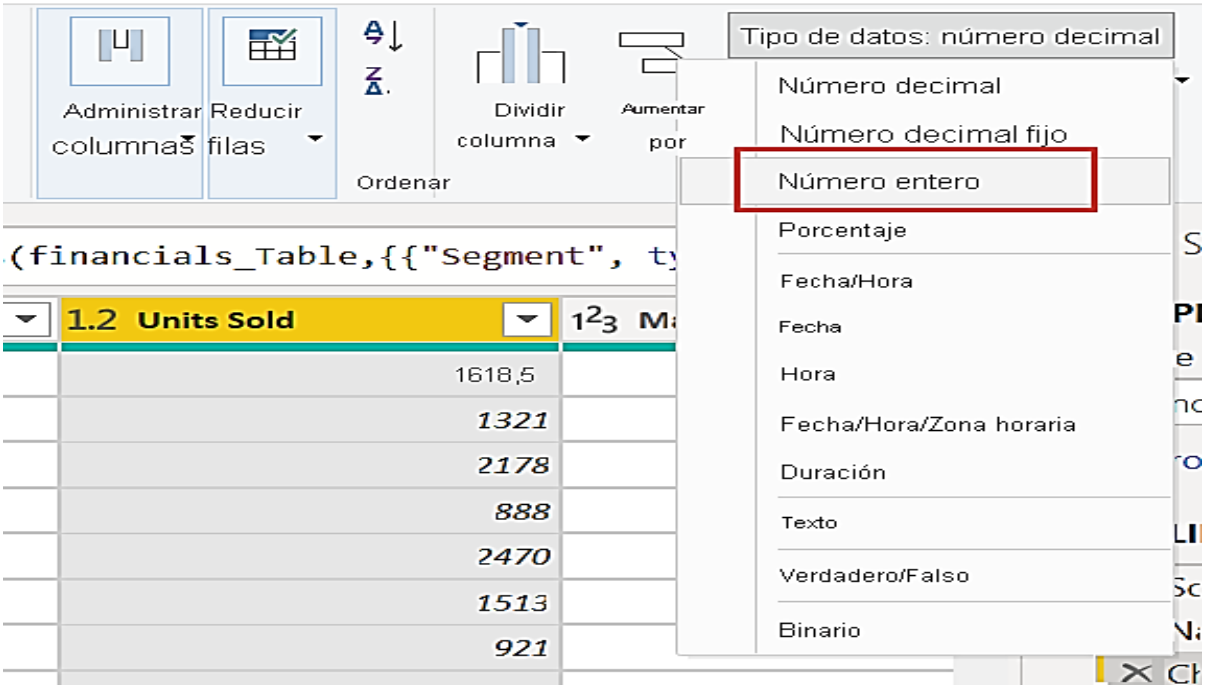
Seleccione la tabla Financials y elija Transformar datos.



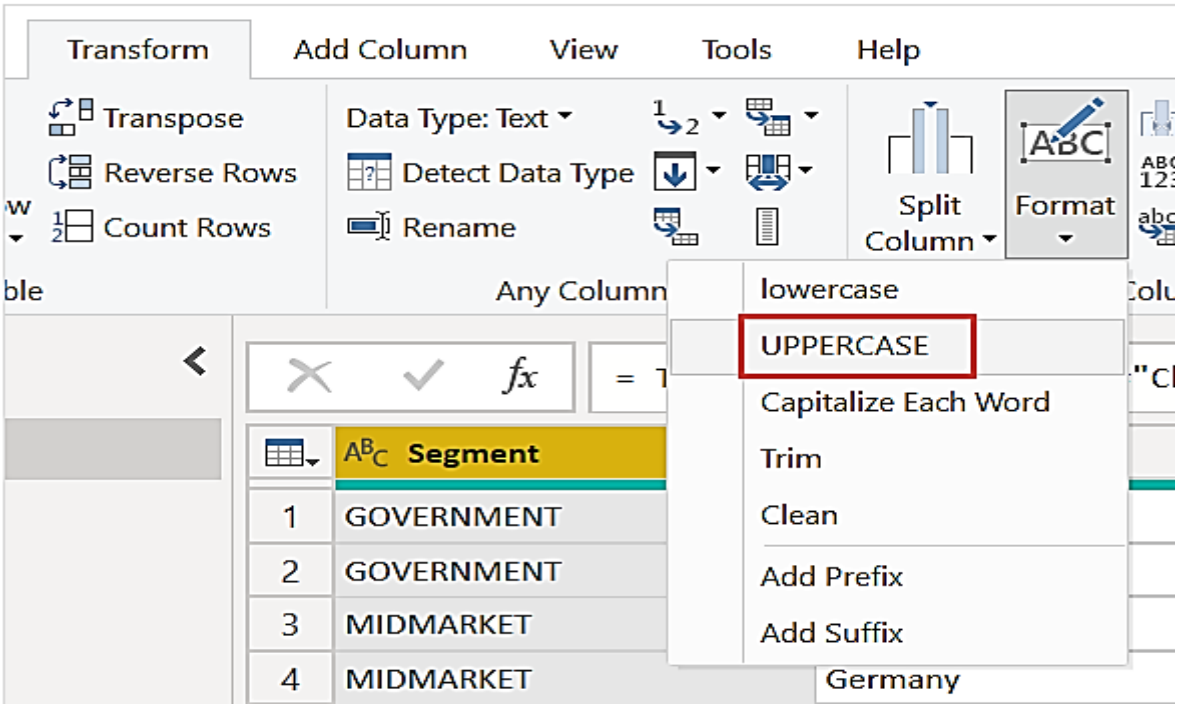
2. Seleccione la columna Unidades vendidas. En la pestaña Transformar datos, seleccione Tipo de datos y, después, Número entero. Elija Sustituir actual para cambiar el tipo de columna.

El principal paso de limpieza de datos que los usuarios realizan con mayor frecuencia es el cambio de los tipos de datos. En este caso, las unidades vendidas tienen formato

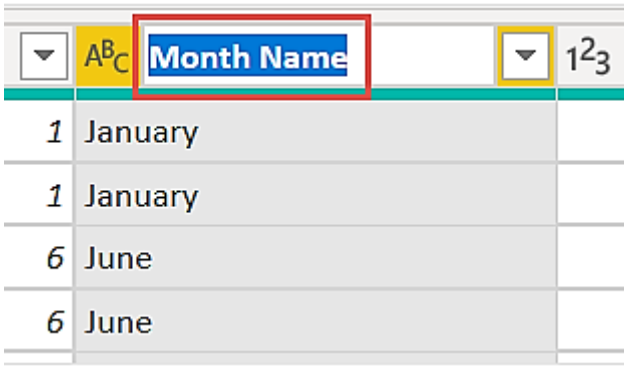
decimal. No tiene sentido tener 0,2 o 0,5 de una unidad vendida, ¿verdad? Por tanto, se cambiará por un número entero.



3. Seleccione la columna Segmento. El objetivo es que los segmentos sean más fáciles de ver en el gráfico más adelante, por lo que se aplicará formato a la columna Segmento. En la pestaña Transformación, seleccione Formato y MAYÚSCULAS.

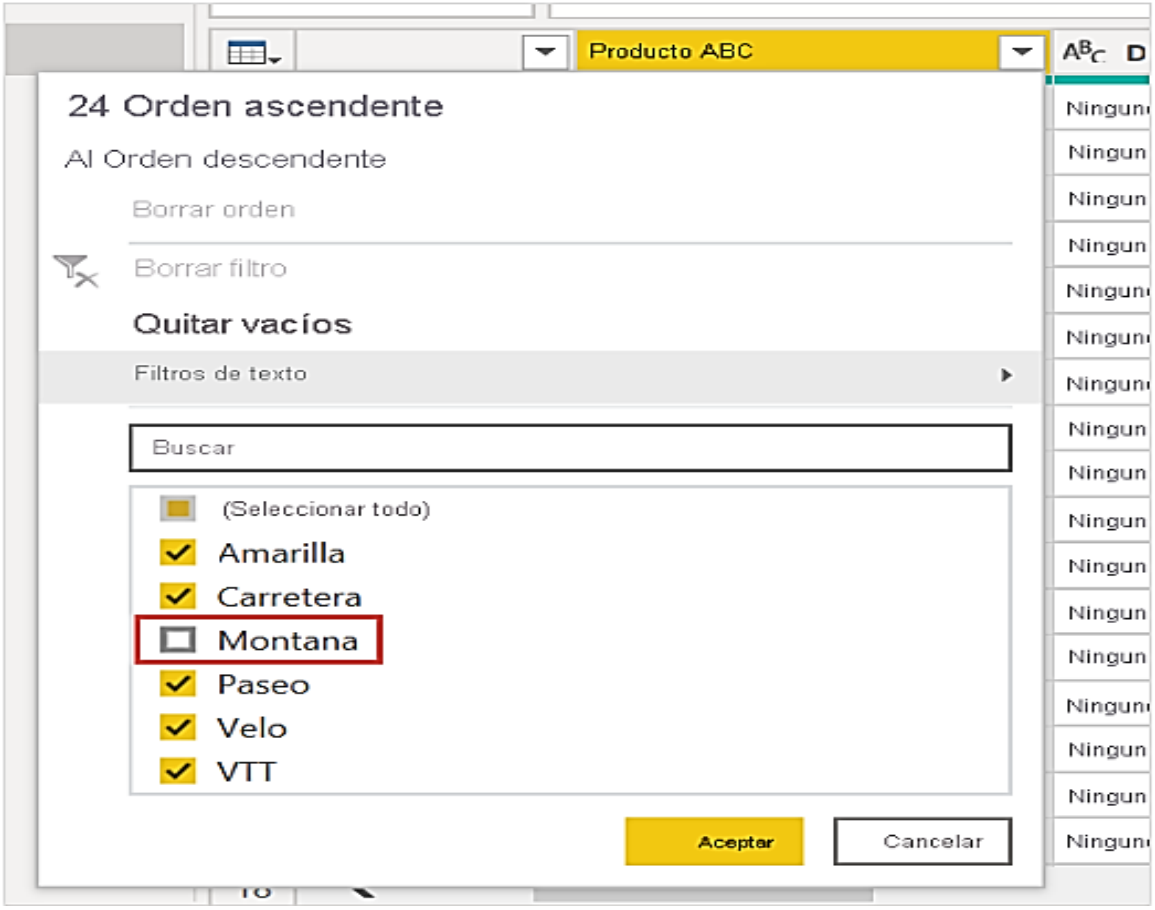


4. Ahora se acortará el nombre de la columna Nombre del mes a solo Mes. Haga doble clic en la columna Nombre del mes y cambie el nombre a solo Mes.

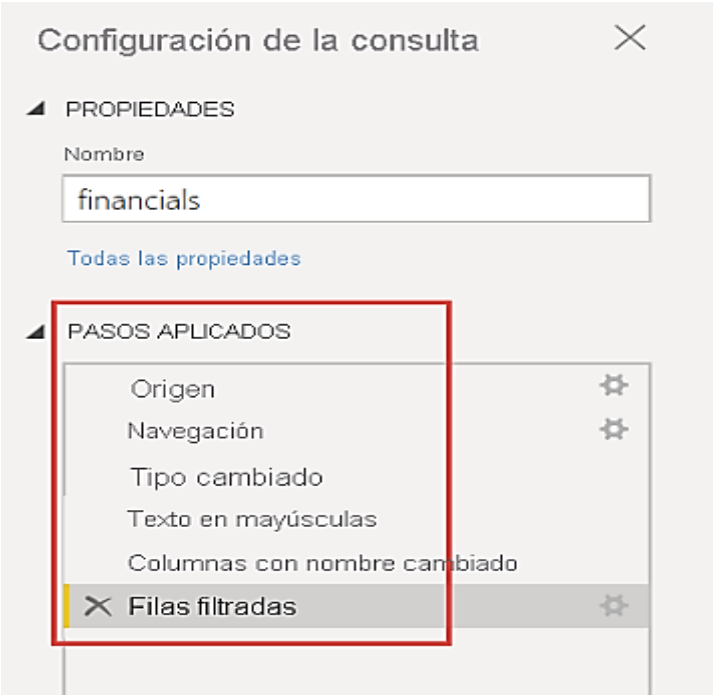


5. En la columna Producto, seleccione la lista desplegable y desactive la casilla situada junto a Montana.

Sabe que el producto Montana dejó de estar disponible el mes pasado, por lo que quiere filtrar estos datos del informe para evitar confusiones.

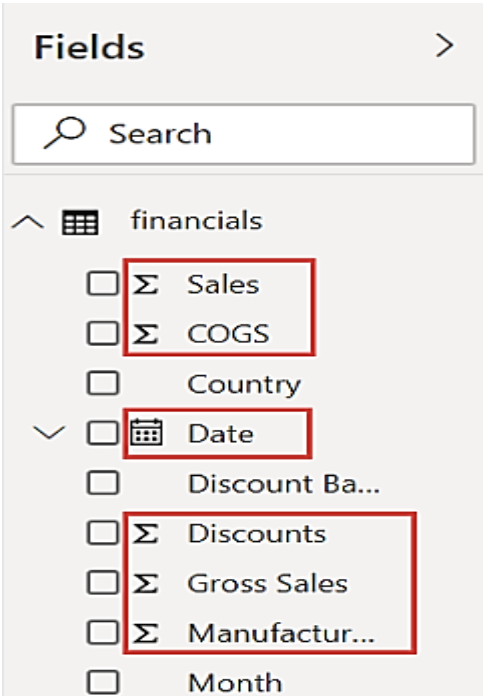


6. Verá que cada transformación se ha agregado a la lista en Configuración de la consulta en Pasos aplicados.



7. De vuelta en la pestaña Inicio, seleccione Cerrar y aplicar. Los datos están casi listos para la creación de un informe.

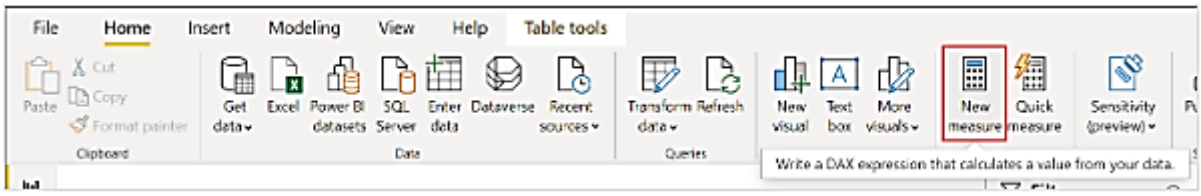
¿Ve el símbolo Sigma en la lista Campos? Power BI ha detectado que esos campos son numéricos. Power BI también indica el campo de fecha con un símbolo de calendario.



Crédito extra: escritura de una expresión en DAX

La escritura de medidas y la creación de tablas en el lenguaje de fórmulas DAX es muy eficaz para el modelado de datos. En la documentación de Power BI hay mucha más información sobre DAX. Por el momento, vamos a escribir una expresión básica y a unir dos tablas.

1. En la cinta de opciones Inicio, seleccione Nueva medida.

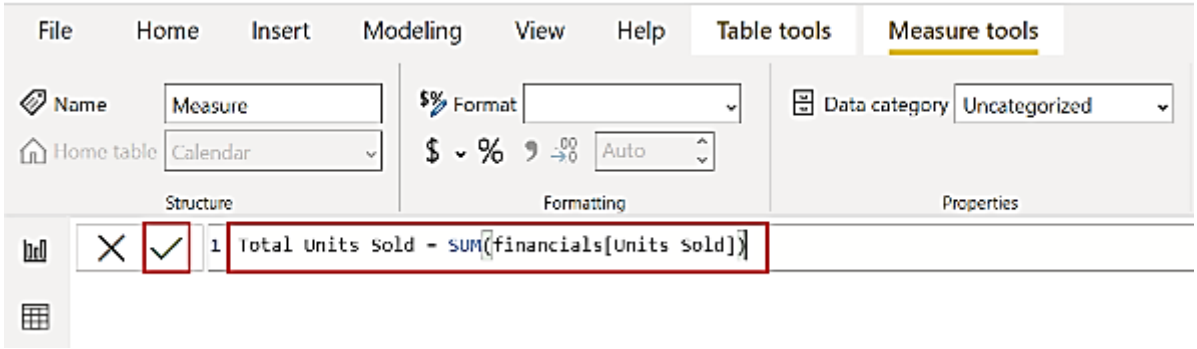


2. Escriba esta expresión para agregar todos los números de la columna Units Sold.

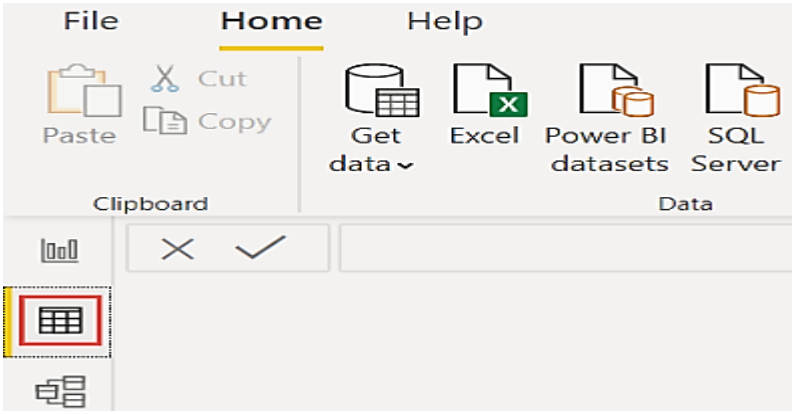
DAX Copiar

```
Total Units Sold = SUM(financials[Units Sold])
```

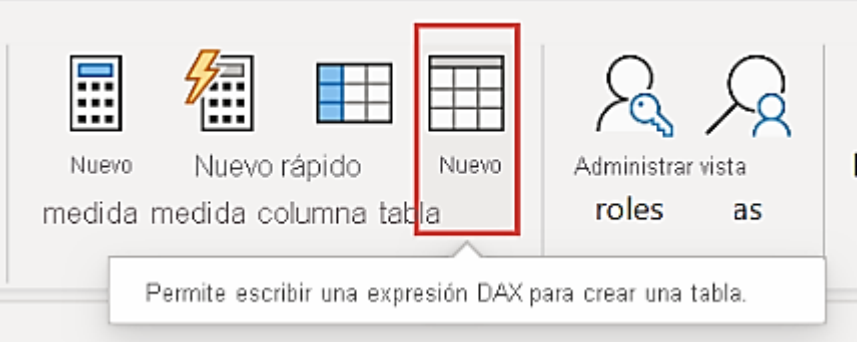
3. Seleccione la marca de verificación para confirmar.



4. Ahora, seleccione la Vista de datos de la izquierda.



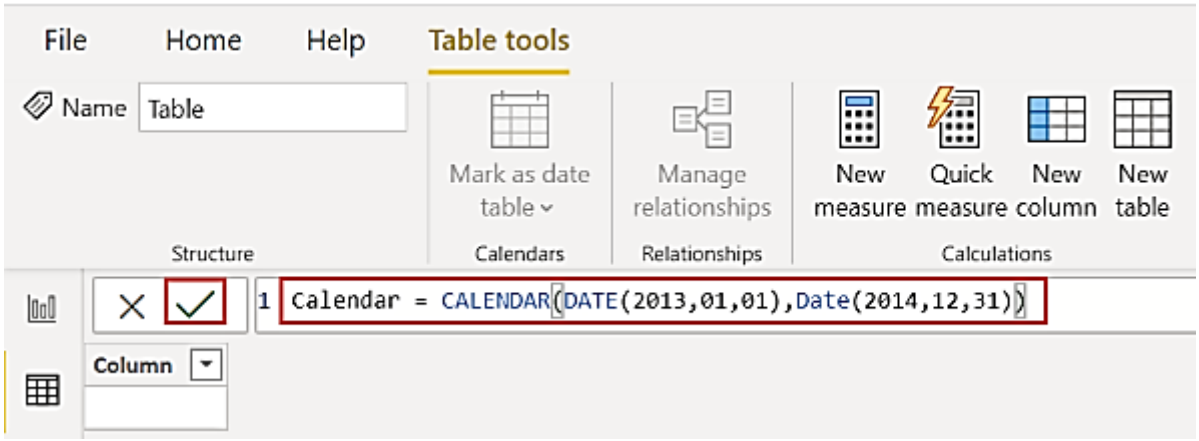
5. En la cinta de opciones Inicio, seleccione Nueva tabla.



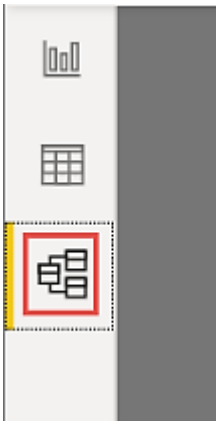
6. Escriba esta expresión para generar una tabla Calendario de todas las fechas entre el 1 de enero de 2013 y el 31 de diciembre de 2014.

```DAX
Calendar = CALENDAR(DATE(2013,01,01),Date(2014,12,31))
```

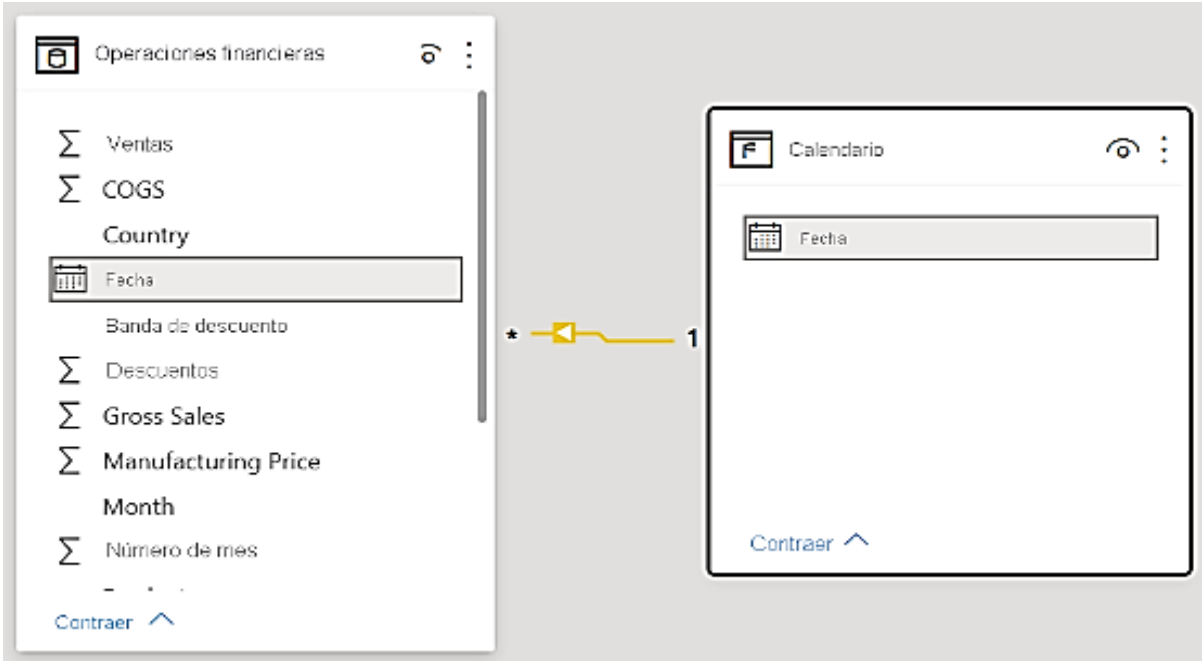
7. Seleccione la marca de verificación para confirmar.



8. Ahora, seleccione Vista de modelo a la izquierda.



9. Arrastre el campo Fecha de la tabla Financials al campo Fecha de la tabla Calendar para combinar las tablas y crear una relación entre ellas.



Crear el informe

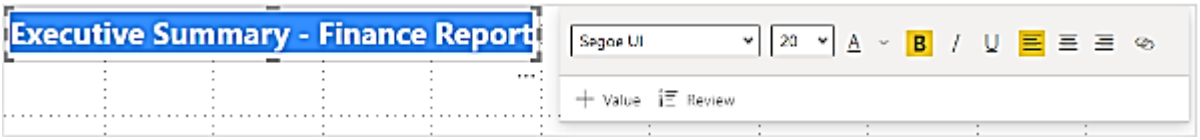
Ahora que ha transformado y cargado los datos, es el momento de crear el informe. En el panel Campos de la derecha, verá los campos del modelo de datos que ha creado. Ahora se creará el informe final, un objeto visual a la vez.



Objeto visual 1: Agregar un título

En la cinta Insertar, seleccione Cuadro de texto. Escriba: Resumen ejecutivo: Informe financiero.

Seleccione el texto que escribió. Establezca el tamaño de fuente en 20 y Negrita.



3. Cambie el tamaño del cuadro para que quepa en una línea.

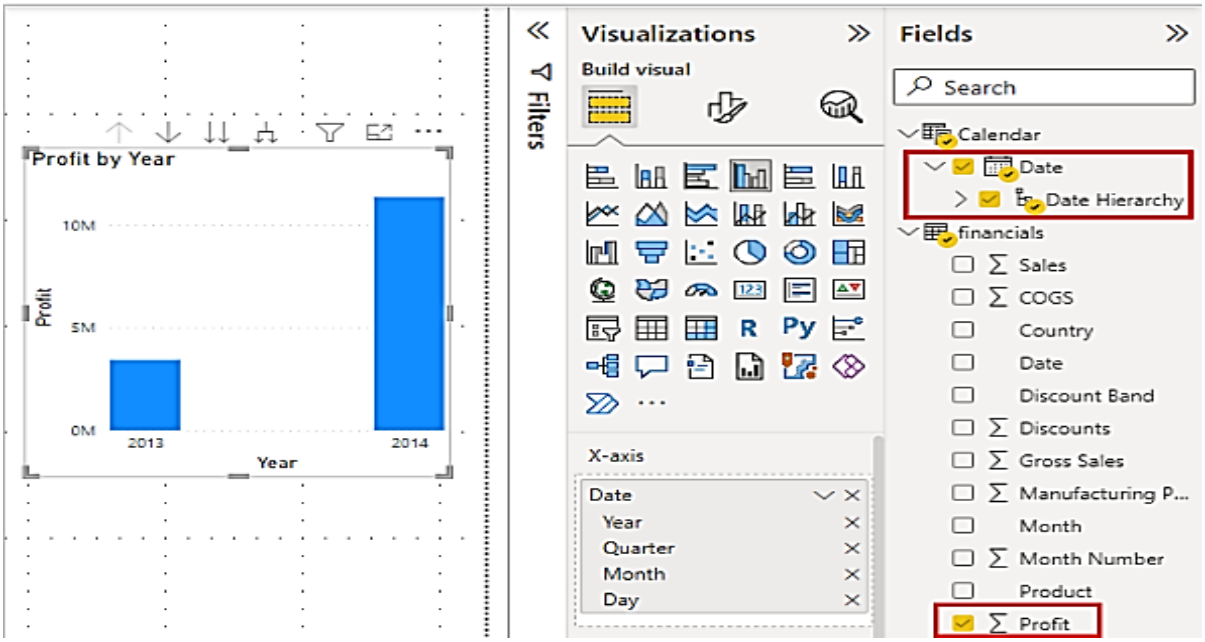
Objeto visual 2: Beneficios por Fecha

Ahora, cree un gráfico de líneas para ver en qué mes y año se ha obtenido el beneficio más alto.

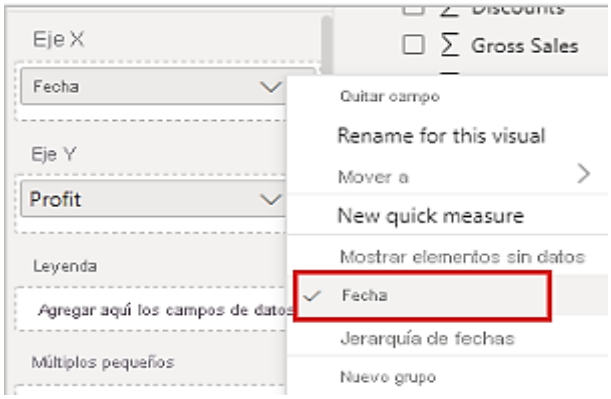
1. Desde el panel Campos, arrastre el campo Beneficios a una zona en blanco del lienzo del informe. De forma predeterminada, Power BI muestra un gráfico de columnas con una columna, Beneficios.

2. Arrastre el campo Fecha al mismo objeto visual. Si creó una tabla Calendario en Crédito adicional: Creación de una tabla en DAX anteriormente en este artículo, arrastre el campo Fecha de la tabla Calendario en su lugar.

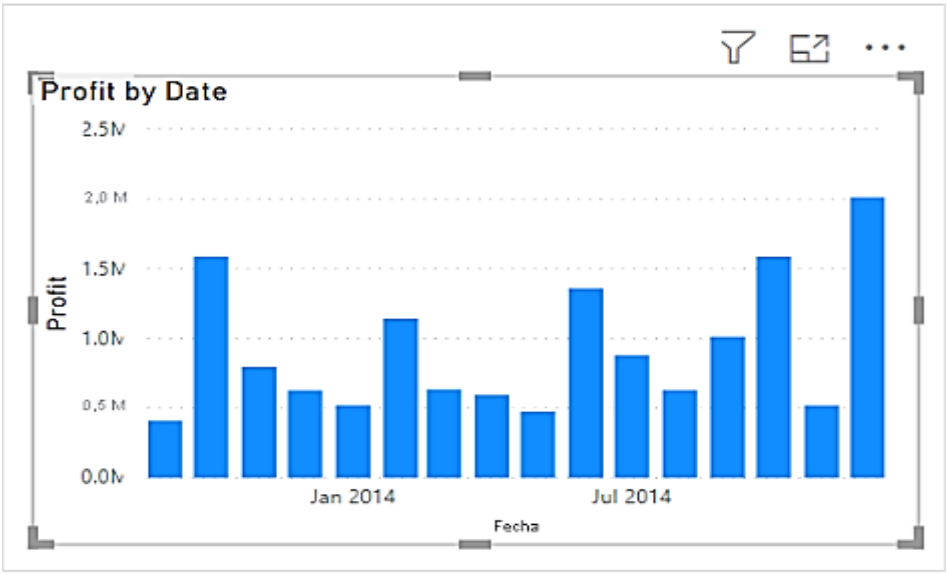
Power BI actualiza el gráfico de columnas para mostrar los beneficios de los dos años.



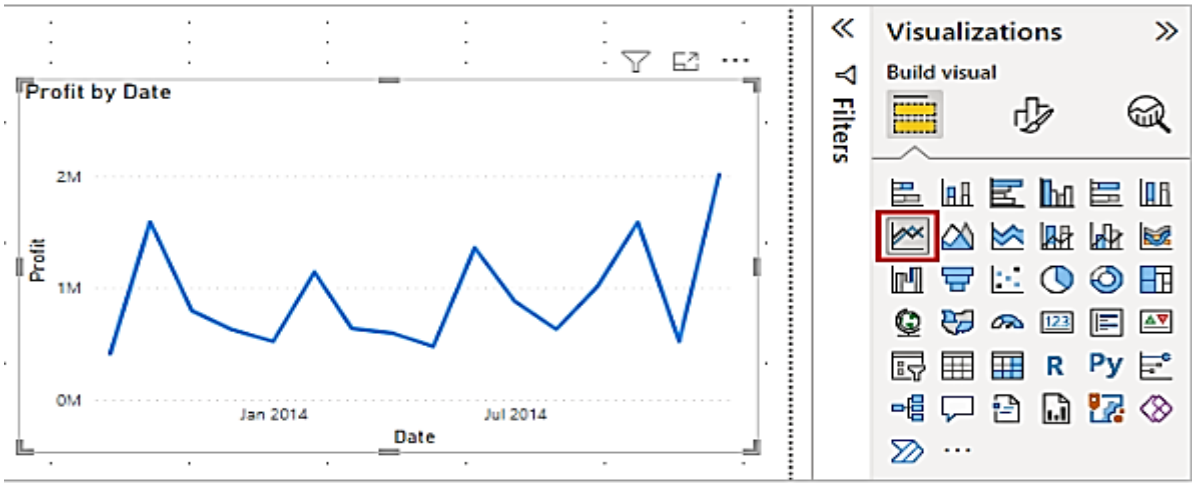
3. En la sección Campos del panel Visualizaciones, seleccione la lista desplegable del valor Eje X. Cambie Fecha de Jerarquía de fechas a Fecha.



Power BI actualiza el gráfico de columnas para mostrar los beneficios de cada mes.



4. En el panel Visualizaciones, cambie el tipo de visualización a Gráfico de líneas.



Ahora puede ver con facilidad que el máximo beneficio se obtuvo en diciembre de 2014.

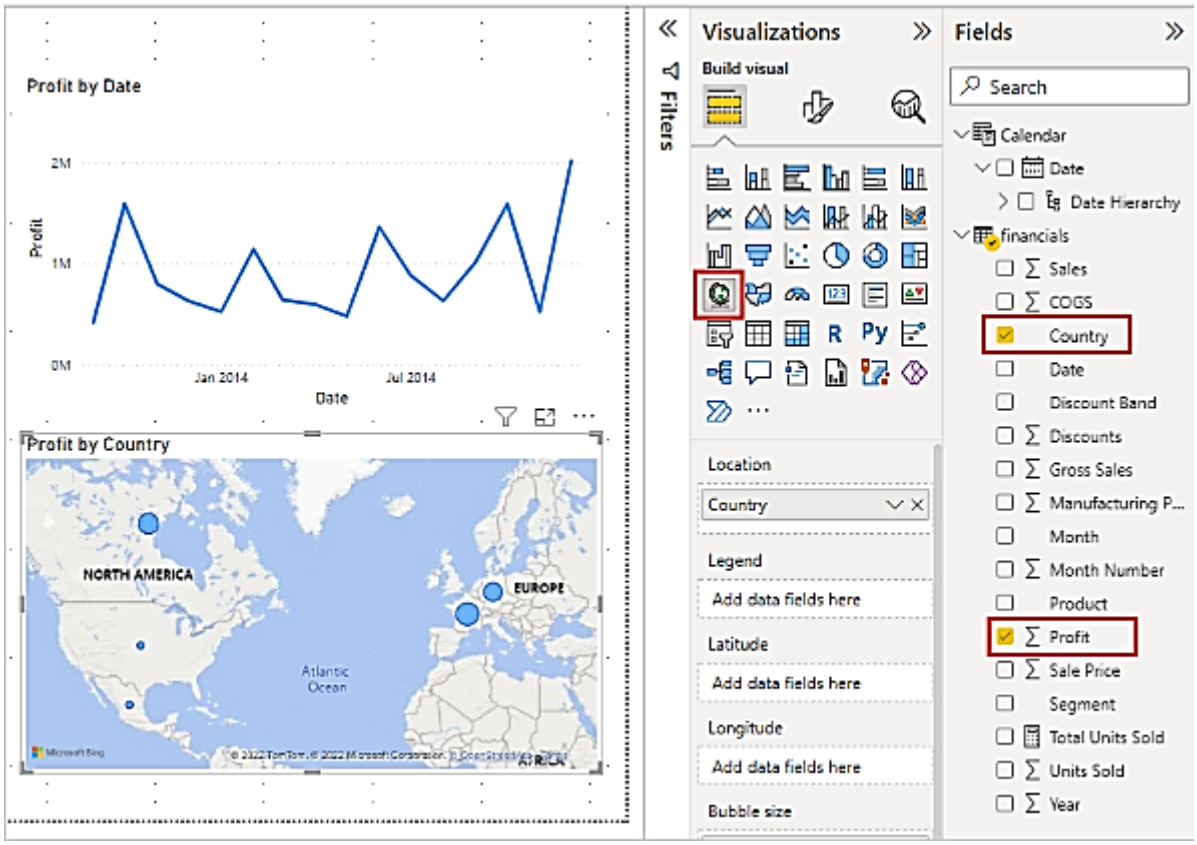
Objeto visual 3: Beneficios por País/Región

Cree un mapa para ver en qué país/región se han obtenido los mayores beneficios.

1. Desde el panel Campos, arrastre el campo País a un área en blanco del lienzo del informe para crear un mapa.

2. Arrastre el campo Beneficios al mapa.

Power BI crea un objeto visual de mapa con burbujas que representan el beneficio relativo de cada ubicación.



Parece que en Europa se obtienen mayores beneficios que en Estados Unidos.

Objeto visual 4: Ventas por Producto y Segmento

Cree un gráfico de barras para determinar en qué empresas y segmentos hay que invertir.

1. Arrastre los dos gráficos que ha creado para situarlos en paralelo en la mitad superior del lienzo. Deje espacio en el lado izquierdo del lienzo.

2. Seleccione un área en blanco en la mitad inferior del lienzo del informe.

3. En el panel Campos, seleccione los campos Ventas, Producto y Segmento.

Power BI crea automáticamente un gráfico de columnas agrupadas.

4. Arrastre el gráfico para que sea lo suficientemente ancho como para rellenar el espacio bajo los dos gráficos superiores.



Parece que la empresa debe seguir invirtiendo en el producto Paseo y dirigirse a los segmentos Pequeña empresa y Administración Pública.

Objeto visual 5: Segmentación de año

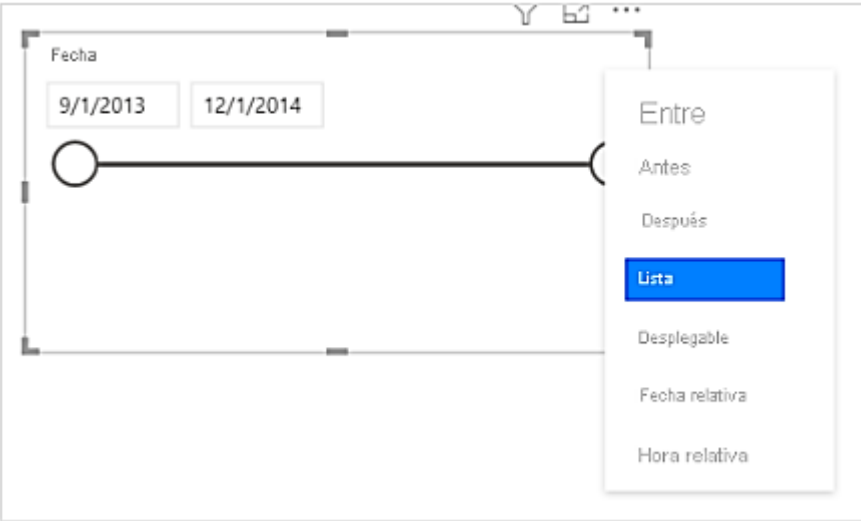
Las segmentaciones son una valiosa herramienta para filtrar los objetos visuales de una página de informe por una selección específica. En este caso, se pueden crear dos segmentaciones que se centren en el rendimiento de cada mes y año. Una segmentación usa el campo de fecha de la tabla original. La otra usa la tabla de fechas que puede haber creado para el crédito adicional anteriormente.

Segmentación de fecha con la tabla original

1. En el panel Campos, seleccione el campo Fecha de la tabla Financials. Arrástrelo al área en blanco a la izquierda del lienzo.

2. En el panel Visualizaciones, seleccione Segmentación.

Power BI crea automáticamente una segmentación de rango numérico.



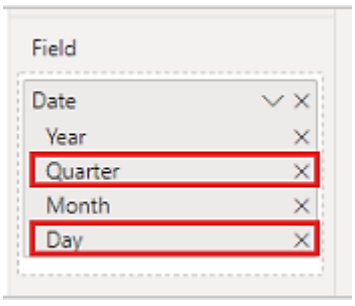
3. Puede arrastrar los extremos para filtrar o seleccionar la flecha situada en la esquina superior derecha y cambiarla a un tipo diferente de segmentación.

Segmentación de fecha con la tabla DAX

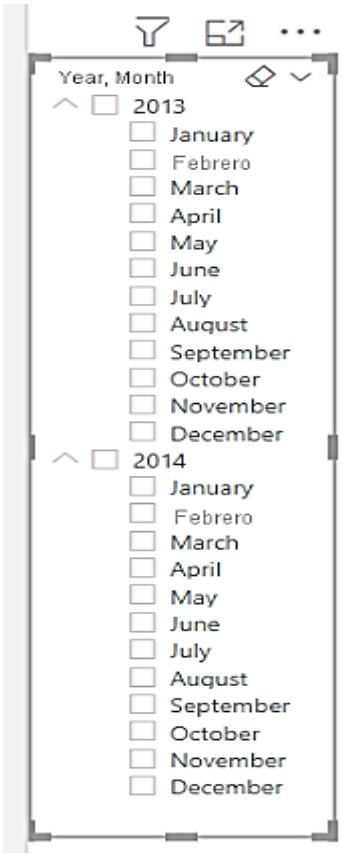
1. En el panel Campos, seleccione el campo Fecha de la tabla Calendario. Arrástrelo al área en blanco a la izquierda del lienzo.

2. En el panel Visualizaciones, seleccione Segmentación.

3. En la sección Campos del panel Visualizaciones, seleccione la lista desplegable de Campos. Quite Quarter (Trimestre) y Day (Día), de modo que solo queden Year (Año) y Month (Mes).



4. Expanda cada año y cambie el tamaño del objeto visual, para que se vean todos los meses.



Usaremos esta segmentación en el informe terminado.

Ahora, si la jefa le pide ver solo los datos de 2013, puede usar la segmentación para seleccionar años o meses concretos de cada año.

Crédito adicional: Formato del informe

Si quiere aplicar un formato ligero a este informe para pulirlo más, puede seguir estos sencillos pasos.

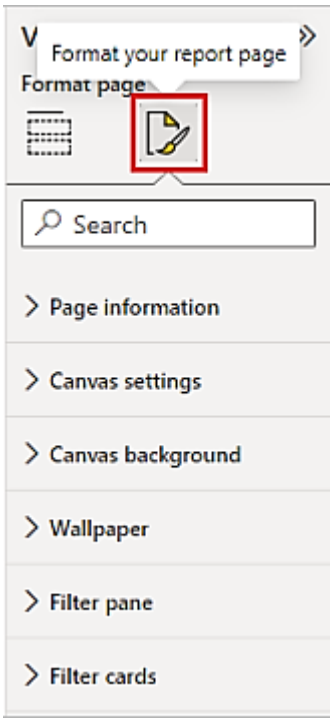
Tema

En la cinta Vista, cambie el tema a Ejecutivo.



Mejora de los objetos visuales

Realice los cambios siguientes en la pestaña Formato del panel Visualizaciones.



1. Seleccione el objeto visual 2. En la sección Título, cambie Texto de título por "Beneficios por mes y año", y Tamaño de texto a 16 pt. Establezca Sombra en Activar.

2. Seleccione el objeto visual 3. En la sección Estilos de mapa, cambie Tema a Escala de grises. En la sección Título, cambie el Tamaño del texto del título a 16 pt. Establezca Sombra en Activar.

3. Seleccione el objeto visual 4. En la sección Título, cambie el Tamaño del texto del título a 16 pt. Establezca Sombra en Activar.

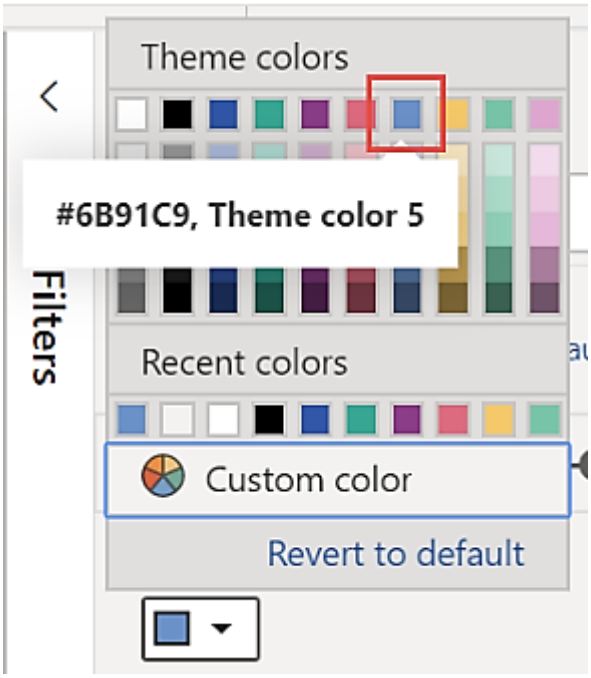
4. Seleccione el objeto visual 5. En la sección Controles de selección, establezca Mostrar opción "Seleccionar todo" en Activar. En la sección Encabezado de segmentación, aumente Tamaño del texto a 16 pt.

Adición de una forma de fondo para el título

1. En la cinta Insertar, seleccione Formas>Rectángulo. Colóquelo en la parte superior de la página y amplíelo para que tenga el ancho de la página y el alto del título.

2. En el panel Formato de forma, en la sección Contorno, cambie Transparencia a 100 %.

3. En la sección Relleno, cambie Color de relleno a Color de tema 5 #6B91C9 (azul).



4. En la pestaña Formato, seleccione Enviar atrás>Enviar al fondo.

5. Seleccione el texto del objeto visual 1, el título, y cambie el color de fuente a Blanco.

Adición de una forma de fondo para los objetos visuales 2 y 3

1. En la cinta Insertar, seleccione Formas>Rectángulo y amplíelo para que tenga el ancho y el alto de los objetos visuales 2 y 3.

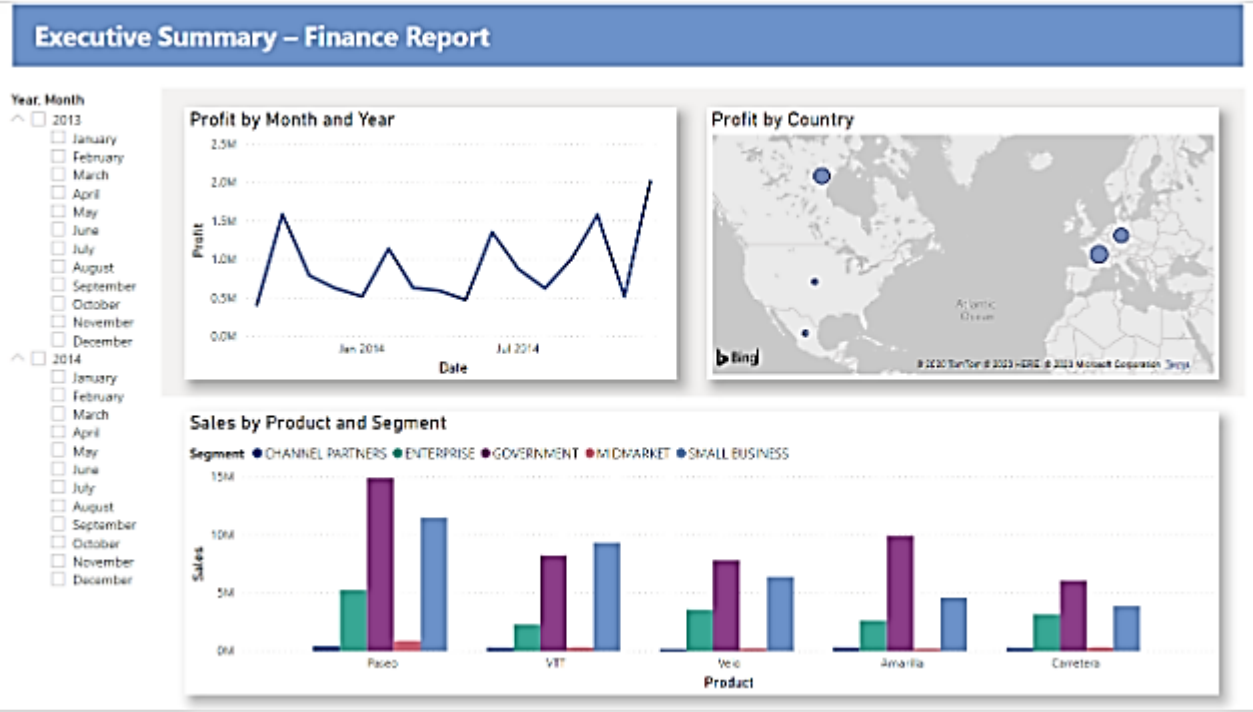
2. En el panel Formato de forma, en la sección Contorno, cambie Transparencia a 100 %.

3. En la sección Relleno, establezca el color en Blanco, 10 % más oscuro.

4. En la pestaña Formato, seleccione Enviar atrás>Enviar al fondo.

Informe finalizado

Este es el aspecto que tendrá el informe acabado:



En resumen, este informe responde a las principales preguntas de la jefa:

Mes y año en los que se ha conseguido el mayor beneficio

Diciembre de 2014

Ubicaciones donde la empresa tiene más éxito (por país/región)

En Europa, concretamente en Francia y Alemania.

Producto y segmento en los que la empresa debe seguir centrando su inversión

La empresa debe seguir invirtiendo en el producto Paseo y dirigirse a los segmentos Pequeña empresa y Administración Pública.

Guardar el informe

En el menú Archivo, seleccione Guardar.

Nota: En caso de ser necesario, se puede compartir el archivo.

GLOSARIO

Lenguaje de programación: Un lenguaje de programación es un lenguaje formal (o artificial, es decir, un lenguaje con reglas gramaticales bien definidas) que proporciona a una persona, en este caso el programador, la capacidad y habilidad de escribir (o programar) una serie de instrucciones o secuencias de órdenes en forma de algoritmos con el fin de controlar el comportamiento de una computadora o sistema informático.

Medida: Es una formulación DAX, que permite obtener respuestas específicas de los campos.

Tabla: En la aplicación del lenguaje DAX, tabla es sinónimo de base de datos.

BIBLIOGRAFÍA

Agreda-Siguenza, A.-J. (s.f.). *Power Bi Transform Model.* Obtenido de SCRIBD: https://es.scribd.com/document/662907355/Power-Bi-Transform-Model

Cárdenas, D. (27 de 04 de 2021). *Funciones DAX en Excel – Descargar PDF.* Obtenido de El Tío Tech: https://eltiotech.com/descargar-funciones-dax-excel-pdf/

Ccance, S. (s.f.). Power BI. En S. Ccance, *Ccance Servicio Integral Informático* (pág. 13).

Escuela De Postgrado Y Educación Continua, Facultad De Ciencias Físicas y Matemáticas, Universidad de Chile. (s.f.). VirtualLabx. En F. D. Escuela De Postgrado Y Educación Continua, *Power BI: Lenguaje Dax y Aplicaciones* (pág. 23). Santiago: Escuela De Postgrado Y Educación Continua, Facultad De Ciencias Físicas y Matemáticas, Universidad de Chile.

G-Talent.net. (2023). *G-Talent.net*. Obtenido de Tutorial para descargar e instalar Power BI: https://www.g-talent.net/blogs/microsoft/descargar-e-instalar-power-bi

Microsoft. (2023). *Docer Argentina.* Obtenido de The Definitive Guide to DAX: Business Intelligence for Microsoft Power BI, SQL Server Analysis Services, and Excel Second Edition (Business Skills). 2nd Edición.: https://docer.com.ar/doc/exv10cn

Quantec. (s.f.). *Power BI - Quantec.* Obtenido de Curso de Power BI: https://www.quantec.mx/_files/ugd/b02660_3f6099e3032a47e888c6505e7d5fb9af.pdf?index=true



INSTITUTO SUPERIOR TECNOLÓGICO UNIVERSITARIO CORPORATIVO

EDWARDS DEMING

Saber qué hacer y cómo hacerlo



Deming

EDITORIAL TECNOLÓGICA

EDICIÓN PRIMERA

ISBN: 978-9942-692-08-5

