



# Introducción a Business Intelligence y Power BI

# DEFINICIÓN Y OBJETIVOS DE BUSINESS INTELLIGENCE (BI)

Business Intelligence (BI) es el proceso de transformar datos en información útil para la toma de decisiones estratégicas.

Implica la integración de herramientas, tecnologías y metodologías que permiten:

- ◆ Recoger datos de múltiples fuentes
- ◆ Organizar y modelar esos datos
- ◆ Visualizarlos de forma comprensible y accionable
- ◆ Facilitar la toma de decisiones basada en datos y no en intuiciones

## Objetivos principales de BI:

- ◆ Mejorar la eficiencia operativa
- ◆ Detectar oportunidades de negocio
- ◆ Optimizar procesos
- ◆ Comprender mejor a clientes y mercados
- ◆ Facilitar el seguimiento y control de KPIs

## 2. ANÁLISIS DE FUENTES DE DATOS

En un entorno BI, la calidad y variedad de las fuentes de datos es clave. En este módulo se analizarán los tipos de datos más comunes y cómo integrarlos en Power BI.

Principales fuentes de datos:



### Archivos locales

Excel  
CSV  
JSON  
XML



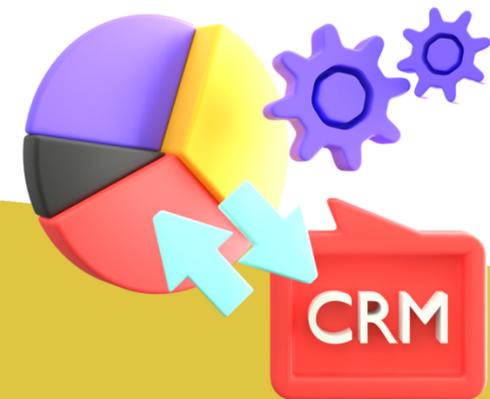
### Bases de datos

SQL Server  
Oracle  
MySQL  
PostgreSQL



### Servicios en la nube

Microsoft SharePoint  
OneDrive  
Azure  
Google Analytics  
Salesforce  
APIs



### Otras aplicaciones empresariales

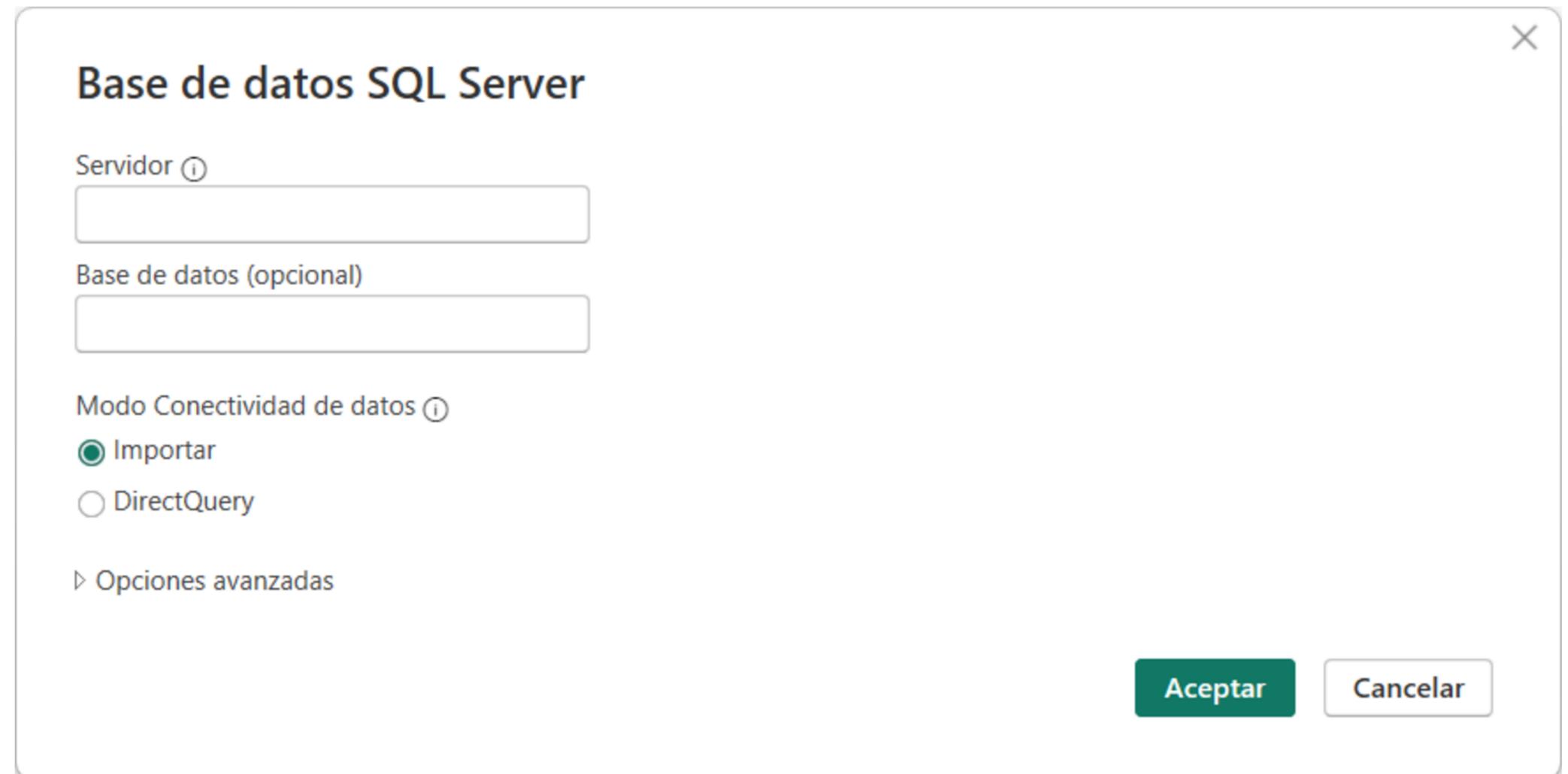
ERP  
CRM  
Herramientas de gestión documental

# Conexión Direct Query vs Import

Cuando conectas Power BI a un origen de datos, puedes elegir cómo traer esos datos:

- Import
- DirectQuery

La elección de uno u otro impacta en el rendimiento, actualización y funcionalidad del informe.



**Base de datos SQL Server**

Servidor ⓘ

Base de datos (opcional)

Modo Conectividad de datos ⓘ

Importar

DirectQuery

▷ Opciones avanzadas

**Aceptar** Cancelar

# Import

## ¿Qué hace?

- Copia los datos del origen a Power BI.
- Los datos se almacenan en el archivo .pbix.

## Características

- ✓ Muy rápido al interactuar (porque trabaja en memoria)
- ✓ Permite usar todas las funciones de DAX
- ✓ Puede crear relaciones complejas
- ✓ Se puede comprimir mucho (optimizando el tamaño del archivo)

## Casos de uso ideales

- ✓ Informes históricos
- ✓ Informes con grandes volúmenes de datos
- ✓ Informes que necesitan máxima interactividad (filtros, segmentaciones)



## Limitaciones

- Los datos no se actualizan en tiempo real
- Necesitas configurar actualizaciones programadas (por ejemplo, cada día o cada hora)

# DIRECT QUERY

¿Qué hace?

- No copia los datos: las consultas se lanzan directamente al origen cada vez que el usuario interactúa con el informe..

## Características

- ✓ Los datos están siempre actualizados (casi en tiempo real)
- ✓ No hay necesidad de programar actualizaciones

## Casos de uso ideales

- ✓ Informes de seguimiento en tiempo real (stock, ventas diarias, sistemas de producción)
- ✓ Cuando no se puede copiar datos por política de seguridad



## Limitaciones

- Depende del rendimiento del sistema origen
- Puede ser más lento al interactuar
- Algunas funciones de DAX y modelado no están disponibles
- Requiere buena configuración de permisos en el origen

## 3. Arquitectura de Power BI



Power BI Cloud Service

Power BI es un ecosistema completo que permite gestionar todo el ciclo de vida del dato. Se compone de varios elementos interrelacionados:

### **Power BI Desktop**

Herramienta de desarrollo local para crear modelos, transformar datos y diseñar visualizaciones.

### **Power BI Service (Cloud)**

Plataforma en la nube para publicar informes, compartir dashboards y colaborar en equipo.

### **Power BI Gateway**

Conector que permite actualizar datos on-premise hacia la nube de

### **Power BI Mobile**

Aplicación para consumir dashboards en dispositivos móviles.

### **Power BI Embedded / API**

Capacidades para integrar visualizaciones de Power BI en aplicaciones personalizadas.

## ¿Por qué Power BI es una excelente elección para empresas?



- Integración nativa con el ecosistema Microsoft
- Lenguaje DAX: potente motor de análisis
- Relación calidad-precio
- Facilidad de uso
- Escalabilidad y flexibilidad
- Actualización constante
- Amplia comunidad y soporte



# **POWER BI** **FUNDAMENTOS** **DE DAX**

# ¿Qué es DAX?

DAX (Data Analysis Expressions) es el lenguaje de fórmulas de Power BI. Permite realizar cálculos avanzados sobre los datos para crear modelos analíticos inteligentes.

No es un lenguaje de programación general: es un lenguaje diseñado específicamente para:

- Crear medidas personalizadas
- Definir columnas calculadas
- Construir tablas calculadas
- Manipular el contexto de los datos
- Implementar lógica de negocio avanzada

# ¿Por qué es importante aprender DAX?

Sin DAX, Power BI sería simplemente una herramienta de visualización. DAX convierte Power BI en una verdadera plataforma de análisis empresarial.

Permite responder preguntas como:

- ¿Cuál es el crecimiento interanual de las ventas?
- ¿Qué porcentaje de clientes han repetido compra este trimestre?
- ¿Cómo varía el margen por producto y región?
- ¿Cuál es el comportamiento de los clientes a lo largo del tiempo?



# ¿DAX es similar a SQL?

**La respuesta es: sí y no.**

DAX y SQL comparten algunos conceptos, pero son lenguajes diseñados para objetivos diferentes y contextos diferentes.

## Similitudes entre DAX y SQL

- Ambos son **lenguajes orientados a datos**: DAX y SQL trabajan con tablas, columnas, filas y relaciones.
- Ambos permiten hacer **filtros, agregaciones y cálculos**: SUM, COUNT, AVG, DISTINCT, etc.
- Ambos se utilizan en entornos de Business Intelligence:
  - **SQL en bases de datos relacionales** (como SQL Server, Oracle, MySQL...)
  - **DAX en modelos tabulares** de Power BI, Analysis Services y Excel Power Pivot.

## Diferencias entre DAX y SQL

## PROPÓSITO



# SQL

- Lenguaje de consulta para recuperar y manipular datos en bases de datos relacionales.
- Se utiliza para consultar, insertar, actualizar y eliminar datos (CRUD).



# DAX

- Lenguaje de expresiones para crear cálculos dentro de un modelo de análisis.
- No se usa para modificar los datos, sino para definir medidas y columnas calculadas que enriquecen el modelo

## Diferencias entre DAX y SQL

## PROPÓSITO



# SQL

- se ejecuta sobre una base de datos relacional (con estructura fija).
- La consulta define el conjunto de datos que se devuelve.



# DAX

- se ejecuta en un modelo en memoria (modelo tabular).
- Su comportamiento depende del contexto de la visualización (filtros aplicados, segmentadores, interacciones del usuario).

# Diferencias entre DAX y SQL

## PROPÓSITO



# SQL

- contexto definido por la propia consulta (SELECT-FROM-WHERE-GROUP BY).



# DAX

- Contexto dinámico:
- Contexto de Fila (cuando se calcula una columna)
- Contexto de Filtro (cuando se calcula una medida en un visual)
- Transiciones de contexto (cuando se utiliza CALCULATE).

## Diferencias entre DAX y SQL

## PROPÓSITO



# SQL

- pensamiento basado en "qué datos quiero devolver".



# DAX

- pensamiento basado en "qué resultado quiero calcular en función del contexto actual del informe".

# **ATAJOS DE TECLADO DAX**

## RESUMEN ATAJO PRINCIPAL A ESCRIBIR CÓDIGO DAX

Nro	Atajo	Acción
1	* Flecha Derecha / Izquierda * ALT + SHIFT + 1	* Expandir / Contraer una tabla * Contraer todas las tablas.
2	* CTRL + SHIFT + L	* Seleccionar y editar simultáneamente todas las ocurrencias de una selección.
3	* CTRL + D	* Seleccionar toda la palabra del cursor * Seleccionar la ocurrencia siguiente de la selección.
4	* ALT + Flecha Arriba / Flecha Abajo	* Mover selección del código hacia arriba o hacia abajo.
5	* ALT + Shift + Flecha Abajo	* Crear una copia de la fila actual debajo.
6	* CTRL + K + C * CTRL + K + U	* Comentar / Descomentar el código seleccionado.
7	* SHIFT + ENTER * CTRL + SHIFT + ENTER	* Insertar una nueva línea encima o debajo del código.
8	* CTRL + SHIFT + K	* Eliminar la fila completa en la que me encuentro.
9	* TAB * SHIFT + TAB	* Insertar / Quitar Sangrías
10	* CTRL + L	* Seleccionar toda una fila



**Training**

SOLUCIONES DE FORMACIÓN