

# ¿Qué es SQL?

**SQL (Structured Query Language) es el lenguaje estándar que se utiliza para comunicarse con bases de datos.**

**Con SQL puedes:**

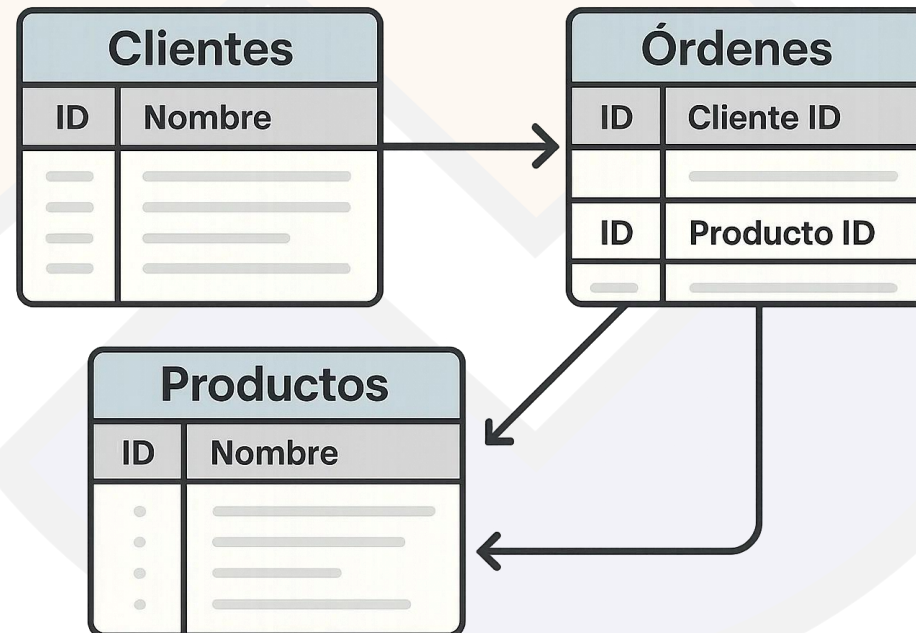
- Crear bases de datos y tablas.
- Insertar, actualizar y eliminar datos.
- Consultar información con filtros.
- Organizar, agrupar y ordenar datos.
- Unir tablas y analizar relaciones.

SQL soporta aproximadamente el 80% de las bases de datos a nivel mundial.

# ¿Qué es una base de datos?

Una **base de datos** es un sistema que permite **guardar y organizar datos** de forma estructurada.

Las bases relacionales (como SQLite o MySQL) se componen de **tablas**, como si fueran **hojas de cálculo** conectadas entre sí.



# ¿Qué es SQLite?

**SQLite es un motor de base de datos liviano, portátil y sin necesidad de servidor.**

## **Ventajas clave de SQLite:**

- No requiere instalación de servidor
- Puedes usarlo directamente en archivos .sqlite o incluso en memoria
- Perfecto para practicar y hacer pruebas
- Muy usado en dispositivos móviles y apps locales

Por eso lo usaremos en este curso: porque funciona perfectamente dentro de Colab, tanto en R como en Python.

# ¿Cuál es la lógica de uso de Python/R con SQL?

## Hay un proceso estandarizado para extraer datos desde bases de datos SQL

1. Conectarse al motor de base de datos (como SQLite):
  - Esto permite que Python o R se comuniquen con una base de datos.
  - En Python se usa, por ejemplo, la librería sqlite3.
  - En R se usa RSQLite y DBI.
2. Ejecutar consultas SQL desde Python o R:
  - Una vez conectados, podemos escribir sentencias SQL (SELECT, FROM, WHERE, etc.) directamente en el código del lenguaje.
3. Recuperar los resultados en un formato de análisis:
  - En Python, los resultados se convierten en un DataFrame con pandas.
  - En R, se convierten en un data.frame o tibble.
  - Esto permite continuar con análisis, visualizaciones o modelos directamente en el lenguaje.

# ¿Por qué conectar Python/R con SQL?

## **SQL es ideal para:**

- Consultar grandes volúmenes de datos rápidamente.
- Filtrar, unir, agrupar y ordenar información directamente en la base de datos.
- Ejecutar operaciones de forma eficiente y segura dentro del servidor.

## **Python y R son ideales para:**

- Realizar análisis estadísticos, machine learning y visualizaciones avanzadas.
- Automatizar flujos de trabajo y crear reportes personalizados.
- Conectarse con múltiples fuentes de datos y herramientas externas.

## **Por eso se combinan:**

- SQL prepara los datos.
- Python/R los analiza, visualiza o modela.
- Juntos forman un flujo de trabajo muy potente y profesional.

# Siguiente clase



Implementaremos SQL con Python y R