

# SQL

El lenguaje de consulta estructurada (SQL-Structured Query Language) es un lenguaje de programación para almacenar y procesar información en una base de datos relacional. Una base de datos relacional almacena información en forma de tabla, con filas y columnas que representan diferentes atributos de datos y las diversas relaciones entre los valores de datos. Puede usar las instrucciones SQL para almacenar, actualizar, eliminar, buscar y recuperar información de la base de datos.

## La historia de SQL

La historia de SQL comienza en 1969, cuando el investigador de IBM Edgar F. Codd definió el modelo de base de datos relacional y la normalización de tablas como requisito indispensable para su funcionamiento. Ese modelo se basa en la asociación de «claves» con varios datos. Por ejemplo, un identificador de usuario se puede asociar con un nombre real y un número de teléfono.

Unos años más tarde, IBM creó un lenguaje para los sistemas de gestión de bases de datos relacionales basado en el trabajo de Codd. Ese lenguaje primero se llamó SEQUEL, siglas de «Structured English Query Language» y después de varias implementaciones y revisiones, pasó a llamarse SQL.

Las pruebas comenzaron en 1978, y luego IBM empezó a desarrollar productos comerciales como SQL/DS en 1981 y DB2 en 1983. Le siguieron otros proveedores, como Sybase, Ingres u Oracle, que lanzó su primer producto en 1979 y se convirtió en el primer proveedor en ofrecer un sistema comercial de administración de bases de datos relacionales SQL.

## Características

Los sistemas de administración de bases de datos relacionales almacenan varias tablas de datos que se relacionan entre sí. MS SQL Server y MySQL son ejemplos de sistemas de administración de bases de datos relacionales.

Los siguientes son los componentes de un sistema de este tipo.

Tabla SQL: es el elemento básico de una base de datos relacional. Las tablas de la base de datos SQL se componen de filas y columnas y al igual que indica el nombre, pueden "relacionarse" entre sí a través de campos denominados "claves".

Instrucciones SQL: o consultas SQL, son instrucciones válidas que los sistemas de administración de bases de datos relacionales entienden. Los elementos del lenguaje SQL son componentes como identificadores, variables y condiciones de búsqueda que forman una instrucción SQL correcta (por ej: Create, Insert etc.).

La implementación del lenguaje de consulta estructurada (SQL) requiere de un servidor que procese las consultas en la base de datos y devuelva los resultados.

Este proceso pasa por varias etapas:

Análisis: se verifica que la instrucción SQL se ajuste a la semántica o reglas de SQL que garantizan la corrección de la instrucción de la consulta. Por ejemplo, el analizador comprueba si el comando SQL termina con punto y coma. Si falta el punto y coma, el analizador devuelve un error. También se valida que el usuario que ejecuta la consulta tenga la autorización necesaria para manipular los datos correspondientes. Por ejemplo, solo los usuarios administradores podrían tener derecho a eliminar datos.

Procesamiento de consultas: se crea un plan para recuperar, escribir o actualizar los datos correspondientes de la manera más eficaz. Por ejemplo, al buscar consultas similares, se vuelve a utilizar métodos de manipulación de datos anteriores o crea uno nuevo.

Ejecución de la instrucción: el motor de la base de datos, es el componente de software que procesa y ejecuta la instrucción SQL aprobada. Lee y recupera los datos en los archivos de la base de datos del almacenamiento en disco físico y devuelve el resultado a la aplicación que lo solicita.

## **Comandos SQL**

Los comandos de lenguaje de consulta estructurada (SQL) son palabras clave o instrucciones SQL específicas que los desarrolladores utilizan para manipular los datos almacenados en una base de datos relacional. Puede clasificar los comandos SQL de la siguiente manera.

-Lenguaje de definición de datos: DDL (Data Definition Language) se refiere a comandos SQL que diseñan la estructura de la base de datos creando y modificando objetos de bases de datos en función de las necesidades empresariales. Por ejemplo, utiliza el comando CREATE para crear objetos de base de datos, como tablas, vistas e índices. Otros ejemplos de comandos DDL son: DROP, ALTER, TRUNCATE, COMMENT, RENAME.

-Lenguaje de consulta de datos: DQL (Data Query Language) consiste en instrucciones para recuperar datos almacenados en bases de datos relacionales. Las aplicaciones de software usan el comando SELECT para filtrar y devolver resultados específicos de una tabla SQL.

-Lenguaje de manipulación de datos: DML (Data manipulation Language)) escribe información nueva o modifica los registros existentes en una base de datos relacional. Por ejemplo, una aplicación usa el comando INSERT para almacenar un nuevo registro en la base de datos. Otros comandos son: UPDATE, DELETE, LOCK, CALL, EXPLAIN PLAN.

-Lenguaje de control de datos: DCL( Data Control Language) es usado para administrar o autorizar el acceso a la base de datos. Por ejemplo, pueden usar el comando GRANT para conceder permisos para que ciertas aplicaciones manipulen una o más tablas o REVOKE para quitarlos.

-Lenguaje de control de transacciones:TCL ( Transaction Control Language) es usado para hacer cambios en la base de datos de manera automática. Por ejemplo, la base de datos usa el

comando ROLLBACK para deshacer una transacción errónea, COMMIT, ROLLBACK, SAVEPOINT, SET TRANSACTION.

### **Estándares de SQL**

Los estándares SQL son un conjunto de pautas definidas formalmente del lenguaje de consulta estructurada (SQL). El Instituto Nacional Estadounidense de Estándares (ANSI) y la Organización Internacional de Normalización (ISO) adoptaron las normas SQL en 1986 que fueron adoptados por los proveedores de software con el fin de crear software de base de datos compatibles entre sí, en el caso de ser necesaria una migración de datos de una base a otra.

MySQL es un sistema de administración de bases de datos relacionales de código abierto ofrecido por Oracle. Los desarrolladores pueden descargar y usar MySQL sin pagar tarifas de licencia. Pueden instalar MySQL en diferentes sistemas operativos o servidores en la nube. MySQL es un sistema de base de datos popular para aplicaciones web. Si bien los comandos SQL están definidos por estándares internacionales, el software MySQL se somete a actualizaciones y mejoras continuas, incorporando instrucciones fuera del estándar ANSI.

### **Ambiente NoSQL**

NoSQL se refiere a bases de datos no relacionales que no usan tablas para almacenar datos. El lenguaje de consulta estructurada (SQL) proporciona un lenguaje de manipulación de datos uniforme, pero la implementación de NoSQL depende de diferentes tecnologías. Los desarrolladores usan SQL para aplicaciones transaccionales y analíticas, mientras que NoSQL es adecuado para aplicaciones con capacidad de respuesta y uso intensivo (Data Mining).

### **Usos del lenguaje SQL**

El lenguaje SQL se usa en todas las áreas en las que se explotan las bases de datos. En otras palabras, se puede encontrar en prácticamente todas las áreas en las que se generan y se analizan datos.

En el sector financiero, las aplicaciones de banca online y de procesamiento de pagos almacenan datos sobre las transacciones financieras y los usuarios. Esos sistemas se basan en bases de datos complejas y, por tanto, requieren el uso de SQL.

Asimismo, las aplicaciones de streaming de música como Spotify hacen un uso intensivo de las bases de datos, por ejemplo, para almacenar los amplios catálogos de archivos de música procedentes de diferentes álbumes de diferentes artistas. SQL permite manipular esos datos para encontrar lo que busca el usuario o incluso almacenar datos sobre el usuario y sus preferencias.

Las redes sociales utilizan SQL para almacenar información sobre el perfil de un usuario, como su biografía o su ubicación. Siempre que un usuario crea una publicación o comparte contenido, SQL permite actualizar la base de datos.

SQL prevalece como el lenguaje de consulta más popular durante las últimas décadas. Esto se debe a sus propiedades:

- Sencillez y comprensibilidad: Para la unificación estructural, usa la sintaxis que es fácil de interpretar en sus comandos.
- Flexibilidad: Los usuarios pueden adaptarlo a cualquier tipo de necesidad, lo que les permite visualizar datos y encontrar soluciones fácilmente.
- Integridad de datos: Protege de fallos y asegura que toda la información sea correcta y esté completa.
- Certificación por ANSI e ISO (International Organization for Standardization): Cuenta con el sello de aprobación de las entidades que garantizan calidad, conformidad y seguridad de su uso.