**Introduccion al SQL**

SQL, que significa Lenguaje de Consulta Estructurado, es un lenguaje diseñado para gestionar datos dentro de sistemas de bases de datos relacionales. A lo largo del tiempo, distintos proveedores han desarrollado sus propias versiones o "sabores" del lenguaje, añadiendo características específicas.  
Nota: Esta definición hace referencia al estándar SQL establecido por ISO/ANSI, y no a una implementación particular de dicho estándar.

**Versiones del estándar SQL**

A lo largo del tiempo, el lenguaje SQL ha pasado por varias actualizaciones oficiales, cada una publicada como parte de los estándares ISO/ANSI. A continuación se presenta un resumen de las principales versiones:

| Año de Publicación | Nombre Corto | Estándar Oficial |
| --- | --- | --- |
| 1986 | SQL-86 | ANSI X3.135-1986, ISO 9075:1987 |
| 1989 | SQL-89 | ANSI X3.135-1989, ISO/IEC 9075:1989 |
| 1992 | SQL-92 | ISO/IEC 9075:1992 |
| 1999 | SQL:1999 | ISO/IEC 9075:1999 |
| 2003 | SQL:2003 | ISO/IEC 9075:2003 |
| 2006 | SQL:2006 | ISO/IEC 9075:2006 |
| 2008 | SQL:2008 | ISO/IEC 9075:2008 |
| 2011 | SQL:2011 | ISO/IEC 9075:2011 |
| 2016 | SQL:2016 | ISO/IEC 9075:2016 |

**Componentes principales del lenguaje SQL**

El lenguaje SQL se divide en tres sublenguajes fundamentales:

Lenguaje de Definición de Datos (DDL): Se utiliza para crear y modificar la estructura de las bases de datos (por ejemplo, tablas, índices, esquemas).

Lenguaje de Manipulación de Datos (DML): Permite realizar operaciones como lectura, inserción, actualización y eliminación de datos. Lenguaje de Control de Datos (DCL): Se usa para gestionar permisos y control de acceso a los datos.

Las operaciones fundamentales del lenguaje SQL para manipular datos se conocen como CRUD, es un acrónimo que representa las **cuatro operaciones fundamentales para gestionar datos** en una base de datos. Crear (Create), Leer (Read), Actualizar (Update) y Eliminar (Delete). Estas acciones se realizan mediante las siguientes instrucciones:

* INSERT para agregar nuevos datos,
* SELECT para consultar o leer datos,
* UPDATE para modificar datos existentes,
* DELETE para eliminar datos.

Además, existe una instrucción más reciente llamada MERGE, que permite realizar combinaciones de las tres operaciones de escritura anteriores (insertar, actualizar y eliminar) en una sola sentencia, lo que la hace muy útil para sincronizar datos entre tablas.

Por otro lado, muchas bases de datos SQL funcionan bajo un modelo cliente/servidor. En este contexto, el término "servidor SQL" se refiere a este tipo de arquitectura, donde un servidor central gestiona las solicitudes de múltiples clientes.

Es importante no confundir este término general con Microsoft SQL Server, que es un producto específico de Microsoft. Aunque utiliza SQL como lenguaje, tiene características propias que se documentan por separado en su propia documentación oficial.

**ALTER TABLE**

El comando ALTER en SQL se utiliza para modificar la estructura de una tabla, ya sea agregando o eliminando columnas, o cambiando restricciones (reglas que controlan los valores permitidos en una columna).

Sintaxis básica

• ALTER TABLE [table\_name] ADD [column\_name] [datatype]

Agregar columnas

ALTER TABLE Employees

ADD StartingDate date NOT NULL DEFAULT GetDate(),

DateOfBirth date NULL

Este comando agrega dos columnas a la tabla Employees:

Eliminar una columna

Este comando elimina por completo la columna salary de la tabla Employees. No solo borra los datos, sino que la columna ya no existirá en la estructura de la tabla.

Eliminar una restricción

Este comando elimina una restricción llamada DefaultSalary de la tabla.  
Nota importante: Antes de eliminar una columna, asegúrate de quitar primero cualquier restricción asociada a ella.

Agregar una restricción

Este comando agrega una restricción llamada DefaultSalary, que establece que el valor por defecto de la columna Salary será 100 si no se especifica otro valor al insertar un nuevo registro

Restricciones en SQL (Constraints)

Las restricciones son reglas que se pueden aplicar a una tabla para asegurar la integridad y validez de los datos. Estas restricciones se pueden definir a nivel de columna o a nivel de tabla.

Tipos de restricciones más comunes:

Clave primaria (Primary Key): Asegura que cada fila en la tabla sea única. No permite valores duplicados ni nulos.

Clave foránea (Foreign Key): Establece una relación con una clave primaria de otra tabla, garantizando la integridad referencial.

No nulo (NOT NULL): Obliga a que una columna siempre tenga un valor (no puede quedar vacía).

Única (UNIQUE): Garantiza que los valores en una columna no se repitan.

valor por defecto (DEFAULT): Asigna automáticamente un valor si no se proporciona uno al insertar datos.

Chequeo (CHECK): Restringe los valores permitidos en una columna, por ejemplo, que un número esté dentro de un rango.

Modificar una columna existente

Este comando cambia el tipo de dato de la columna StartingDate de DATE a DATETIME, obliga a que no pueda estar vacía (NOT NULL) y le asigna como valor por defecto la fecha y hora actual (GETDATE()).

Agregar una clave primaria

Este comando establece que la columna ID será la clave primaria de la tabla Employees.

Si deseas que la clave primaria esté compuesta por más de una columna (por ejemplo, ID y FName), puedes hacerlo así:

Esto se conoce como clave primaria compuesta, y sirve para identificar de forma única cada fila usando más de una columna.

### ✅ 1. ¿Qué es SQL?

SQL es un lenguaje estándar para interactuar con **bases de datos relacionales**. Se usa para **crear, modificar, consultar y administrar** los datos almacenados en ellas.

### ✅ 2. ¿Qué es una base de datos relacional?

Una base de datos relacional almacena datos en **tablas** (también llamadas relaciones), que consisten en filas y columnas, similares a una hoja de cálculo. Cada tabla representa una entidad, y las filas son instancias de esa entidad.

### ✅ 3. Principales operaciones en SQL (CRUD)

| **Operación** | **SQL asociada** | **Descripción** |
| --- | --- | --- |
| Create | INSERT | Agrega nuevos datos a una tabla |
| Read | SELECT | Consulta datos desde una tabla |
| Update | UPDATE | Modifica datos existentes |
| Delete | DELETE | Elimina datos |

### ✅ 4. Tipos de comandos SQL

1. **DDL (Data Definition Language)**
   * CREATE, ALTER, DROP  
     Define y modifica la estructura de la base de datos.
2. **DML (Data Manipulation Language)**
   * SELECT, INSERT, UPDATE, DELETE  
     Manipula los datos dentro de las tablas.
3. **DCL (Data Control Language)**
   * GRANT, REVOKE  
     Controla permisos y accesos a los datos.
4. **TCL (Transaction Control Language)**
   * COMMIT, ROLLBACK, SAVEPOINT  
     Gestiona transacciones para asegurar la integridad de los datos.

### ✅ 5. Claves y relaciones

* **Clave primaria (PRIMARY KEY)**: Identifica de forma única cada fila en una tabla.
* **Clave foránea (FOREIGN KEY)**: Define una relación entre dos tablas.
* **Integridad referencial**: Garantiza que las relaciones entre tablas sean válidas y consistentes.

### ✅ 6. Consultas SQL básicas

* **Filtrar datos**: SELECT \* FROM empleados WHERE salario > 3000;
* **Ordenar resultados**: ORDER BY, DESC / ASC
* **Agrupar y resumir**: GROUP BY, SUM(), COUNT()
* **Unir tablas**: JOIN (INNER, LEFT, RIGHT, FULL)

### ✅ 7. Ventajas del uso de SQL

* Estándar ampliamente adoptado.
* Facilita la gestión y consulta de grandes volúmenes de datos.
* Potente y flexible para realizar análisis, reportes y mantenimiento de información.

### ✅ 8. Conceptos avanzados relacionados (introductoriamente)

* **Índices**: mejoran el rendimiento de las consultas.
* **Vistas**: consultas guardadas como objetos reutilizables.
* **Procedimientos almacenados y triggers**: lógica del negocio dentro del motor de base de datos.
* **Normalización**: proceso de diseño de bases de datos que minimiza la redundancia.

¿Quieres que te prepare una guía visual, una hoja de trucos o ejemplos de cada concepto?

4o

##### Tú dijiste:

podrias dar mas detalles del punto 3 y 4

###### ChatGPT Plus

¡Claro! A continuación te explico con más detalle los **puntos 3 y 4** sobre las operaciones CRUD y los tipos de comandos SQL:

## ✅ 3. Operaciones CRUD en SQL (Create, Read, Update, Delete)

CRUD es un acrónimo que representa las **cuatro operaciones fundamentales para gestionar datos** en una base de datos.

### 🔹 a) CREATE – Insertar datos (INSERT)

Permite **agregar nuevas filas** a una tabla existente.

**Ejemplo:**

sql

CopiarEditar

INSERT INTO empleados (nombre, edad, salario)

VALUES ('Juan Pérez', 30, 2500);

### 🔹 b) READ – Consultar datos (SELECT)

Permite **leer o consultar datos** desde una o varias tablas.

**Ejemplo:**

sql

CopiarEditar

SELECT nombre, salario

FROM empleados

WHERE salario > 2000;

### 🔹 c) UPDATE – Modificar datos

Permite **actualizar valores existentes** en una tabla.

**Ejemplo:**

sql

CopiarEditar

UPDATE empleados

SET salario = salario + 500

WHERE nombre = 'Juan Pérez';

### 🔹 d) DELETE – Eliminar datos

Permite **eliminar una o varias filas** de una tabla.

**Ejemplo:**

sql

CopiarEditar

DELETE FROM empleados

WHERE edad > 65;

⚠️ Cuidado: Si no se usa WHERE, eliminarás **todas** las filas de la tabla.

## ✅ 4. Tipos de comandos SQL

SQL está dividido en varios subconjuntos de comandos, cada uno con una finalidad diferente:

### 🔸 1. ****DDL (Data Definition Language)**** – Lenguaje de definición de datos

Sirve para **definir y modificar la estructura** de la base de datos (tablas, columnas, índices).

| **Comando** | **Función** |
| --- | --- |
| CREATE | Crea tablas, bases de datos, vistas, índices |
| ALTER | Modifica estructuras existentes (añadir/eliminar columnas) |
| DROP | Elimina estructuras de la base de datos |
| TRUNCATE | Borra todos los datos de una tabla sin dejar rastros de transacción |

**Ejemplo:**

sql

CopiarEditar

CREATE TABLE empleados (

id INT PRIMARY KEY,

nombre VARCHAR(100),

edad INT,

salario DECIMAL(10,2)

);

### 🔸 2. ****DML (Data Manipulation Language)**** – Lenguaje de manipulación de datos

Permite **trabajar con los datos** dentro de las estructuras definidas (tablas).

| **Comando** | **Función** |
| --- | --- |
| SELECT | Consulta datos |
| INSERT | Agrega datos |
| UPDATE | Modifica datos |
| DELETE | Elimina datos |

Ya vimos ejemplos de cada uno en el punto anterior (CRUD).

### 🔸 3. ****DCL (Data Control Language)**** – Lenguaje de control de datos

Administra los **permisos de acceso** a los datos.

| **Comando** | **Función** |
| --- | --- |
| GRANT | Otorga privilegios a usuarios |
| REVOKE | Revoca (quita) privilegios otorgados |

**Ejemplo:**

sql

CopiarEditar

GRANT SELECT, INSERT ON empleados TO usuario1;

### 🔸 4. ****TCL (Transaction Control Language)**** – Lenguaje de control de transacciones

Controla cómo se **confirman o deshacen los cambios** en la base de datos. Ideal para garantizar la integridad de los datos.

| **Comando** | **Función** |
| --- | --- |
| COMMIT | Confirma los cambios realizados en una transacción |
| ROLLBACK | Revierte los cambios no confirmados |
| SAVEPOINT | Crea un punto intermedio dentro de una transacción para hacer rollback parcial |

**Ejemplo:**

sql

CopiarEditar

BEGIN;

UPDATE empleados SET salario = salario + 1000 WHERE id = 1;

SAVEPOINT antes\_del\_aumento;

UPDATE empleados SET salario = salario - 500 WHERE id = 2;

ROLLBACK TO antes\_del\_aumento;

COMMIT;