

1. ¿Qué es SQL?

SQL (Structured Query Language o *Lenguaje de Consulta Estructurado*) es un lenguaje que se utiliza para **trabajar con bases de datos**.

SQL sirve para **hablar con una base de datos** y decirle qué hacer.

2. ¿Para qué se utiliza SQL?

SQL se usa principalmente para:

a. Consultar datos

Buscar información dentro de una base de datos.

```
SELECT * FROM Autores;
```

Tarea que realiza este código: **Muestra todos los registros de la tabla.**

2. Insertar datos

Vamos adicionar o Agregar información nueva en nuestra tabla autores, debemos utilizar este código SQL:

```
INSERT INTO Autores (Nombre, Apellidos)  
VALUES ('Gabriel', 'García Márquez');
```

3. Actualizar datos

Modificar información existente en la tabla autores: Emplear el siguiente código SQL:

```
UPDATE Autores  
SET Nombre = 'Gabo'  
WHERE Codigo = AAA10;
```

4. Eliminar datos

Borrar o eliminar registros de nuestra base de datos llamada autores.

```
DELETE FROM Autores  
WHERE Codigo = AAA12;
```

5. Para crear estructuras o estructura de una base de datos se puede emplear el siguiente código SQL

Crear tablas o bases de datos.

```
CREATE TABLE Autores (  
  Codigo INT,  
  Nombre TEXT,  
  Apellidos TEXT  
);
```

3. Estructura básica de SQL (Sintaxis)

La sintaxis de SQL sigue un orden lógico. La más importante es la consulta SELECT:

```
SELECT campos  
FROM tabla  
WHERE condición  
ORDER BY campo;
```

Ejemplo explicado paso a paso de nuestra tabla llamada autores:

```
SELECT Nombre, Apellidos  
FROM Autores  
WHERE Codigo > 5  
ORDER BY Nombre;
```

SIGNIFICADO DE CADA UNA DE LAS INSTRUCCIONES:

- ✓ SELECT → Qué quiero ver
- ✓ FROM → De dónde lo saco
- ✓ WHERE → Condición
- ✓ ORDER BY → Cómo lo ordeno (alfabéticamente ascendente, descendente, numérico mayor a menor, menor a mayor).

4. Tipos de comandos SQL

SQL se divide en varios tipos:

1. DQL (Data Query Language)

Consultar datos

- SELECT

2. DML (Data Manipulation Language)

Modificar datos

- INSERT
- UPDATE
- DELETE

3. DDL (Data Definition Language)

Definir estructuras

- CREATE
- ALTER
- DROP

4. DCL (Control de acceso)

Permisos

- GRANT
- REVOKE

5. ¿Cómo se usa SQL en Microsoft Access?

Microsoft Access es un gestor de bases de datos que **permite usar SQL de forma visual y manual.**

Paso a paso para usar SQL en Access:

Paso 1: Abrir la base de datos y la tabla autores:

Abre tu archivo de Access.

Paso 2: Crear una consulta

1. Ve a **Crear**
2. Haz clic en **Diseño de consulta**
3. Cierra la ventana de tablas (si aparece)

Paso 3: Cambiar a vista SQL

1. En la parte superior de access, haz clic en:
Vista SQL
2. **Paso 4: Escribir código SQL**

Ejemplo:

```
SELECT Nombre, Apellidos  
FROM Autores  
WHERE Codigo > AAA24;
```

Paso 5: Ejecutar la consulta

Haz clic en **Ejecutar** (↵)

Verás los resultados inmediatamente, explique la información que observa en la pantalla del computador.

6. Características de SQL en Access

- ✓ Usa una versión llamada **SQL de Access (Jet SQL o ACE SQL)**
- ✓ Es más simple que otros SQL como MySQL o SQL Server
- ✓ Integra herramientas visuales (no necesitas escribir todo el código)
- ✓ Permite consultas combinadas con formularios e informes

7. Ejemplos que el estudiante debe verificar en Access y entregar para evaluación:

Buscar autores con premios de nuestra base de datos y en la tabla autores:

```
SELECT Nombre, Apellidos  
FROM Autores  
WHERE [Premios/Reconocimientos] IS NOT NULL;
```

Ordenar por apellido ejercicio No. 1

```
SELECT *  
FROM Autores  
ORDER BY Apellidos ASC;
```

Buscar por nombre específico ejemplo No. 2

```
SELECT *  
FROM Autores  
WHERE Nombre = 'Gabriel';
```

8. Ventajas de SQL

- ✓ Fácil de aprender
- ✓ Muy poderoso
- ✓ Funciona en casi todos los sistemas
- ✓ Permite automatizar consultas

9. Conceptos clave que debes recordar

- SQL = lenguaje para bases de datos
- SELECT = consultar
- INSERT = agregar
- UPDATE = modificar
- DELETE = eliminar

- Access permite usar SQL fácilmente

10. TRABAJO FINAL

SQL es el lenguaje que te permite **controlar completamente una base de datos**. En Access, puedes usarlo tanto **de forma visual como escribiendo código**, lo que lo hace ideal para principiantes y avanzados.

RESUMEN DEL TRABAJO REALIZADO HASTA LA FECHA

ESTRUCTURA BASE DE LA TABLA

Antes de comenzar, asumimos que tienes esta tabla:

Autores

- Código (Numérico)
- Nombre (Texto)
- Apellidos (Texto)
- Premios/Reconocimientos (Texto)
- Biografía (Texto)
- Imagen/Foto (Texto o Ruta)
- Audio (Texto o Ruta)

ACTIVIDAD No. 1

Usted debe demostrar el funcionamiento correcto de cada uno de los siguientes códigos SQL, haciendo uso de su base de datos y de la tabla autores.

1. Mostrar todos los autores

Objetivo: ver toda la información contenida en la tabla autores:

```
SELECT * FROM Autores;
```

2. Mostrar solo nombres y apellidos de la tabla autores:

```
SELECT Nombre, Apellidos FROM Autores;
```

3. Ordenar autores por apellido

```
SELECT * FROM Autores  
ORDER BY Apellidos ASC;
```

4. Buscar autores con código mayor a 5

```
SELECT * FROM Autores  
WHERE Codigo > BBB34;
```

5. Buscar autores con nombre específico

```
SELECT * FROM Autores  
WHERE Nombre = 'Gabriel';
```

6. Mostrar autores sin premios

```
SELECT * FROM Autores  
WHERE [Premios/Reconocimientos] IS NULL;
```

7. Mostrar autores con premios

```
SELECT Nombre, Apellidos  
FROM Autores  
WHERE [Premios/Reconocimientos] IS NOT NULL;
```

ACTIVIDAD No. 2

Verificar cada uno de los siguientes ejercicios utilizando su base de datos y la tabla autores.

8. Buscar autores cuyo nombre empiece por “A”

```
SELECT * FROM Autores  
WHERE Nombre LIKE 'A*';
```

9. Buscar autores cuyo apellido termine en “ez”

```
SELECT * FROM Autores  
WHERE Apellidos LIKE '*ez';
```

10. Contar número total de autores

```
SELECT COUNT(*) AS TotalAutores  
FROM Autores;
```

11. Contar autores con premios

```
SELECT COUNT(*) AS AutoresConPremios  
FROM Autores  
WHERE [Premios/Reconocimientos] IS NOT NULL;
```

12. Ordenar por nombre descendente

```
SELECT Nombre, Apellidos  
FROM Autores  
ORDER BY Nombre DESC;
```

13. Mostrar autores entre dos códigos

```
SELECT *  
FROM Autores  
WHERE Codigo BETWEEN AAA10 AND AAA15;
```

14. Crear un alias de campo

```
SELECT Nombre AS NombreAutor, Apellidos AS ApellidoAutor  
FROM Autores;
```

ACTIVIDAD No. 3

Verificar y comprobar cada uno de los siguientes ejercicios con la base de datos y tabla autores construidas el trimestre anterior.

15. Insertar un nuevo autor:

```
INSERT INTO Autores (Codigo, Nombre, Apellidos)  
VALUES (20, 'Jorge', 'Rengifo Gutiérrez');
```

16. Actualizar nombre de un autor

```
UPDATE Autores  
SET Nombre = 'Gabo'  
WHERE Codigo = 1;
```

17. Eliminar un autor

```
DELETE FROM Autores  
WHERE Codigo = 20;
```

18. Mostrar longitud del nombre

```
SELECT Nombre, LEN(Nombre) AS LongitudNombre  
FROM Autores;
```

19. Convertir nombres a mayúsculas

```
SELECT UCASE(Nombre) AS NombreMayuscula  
FROM Autores;
```

20. Concatenar nombre completo

```
SELECT Nombre & ' ' & Apellidos AS NombreCompleto  
FROM Autores;
```

RECOMENDACIONES PARA PRACTICAR

- ✓ Ejecuta cada consulta en **Vista SQL**
- ✓ Modifica los valores para experimentar
- ✓ Combina consultas (por ejemplo: WHERE + ORDER BY)
- ✓ Crea tus propios datos en la tabla.