# Sentencias SQL para Modificar los Registros en una Base de Datos

**INSERT – Insertar nuevos registros**

Se usa para agregar filas a una tabla.

📌 **Ejemplo: Insertar un solo registro**

**INSERT** **INTO** empleados (nombre, edad, puesto, salario)

**VALUES** ('Juan Pérez', 30, 'Desarrollador', 50000);

📌 **Ejemplo: Insertar múltiples registros**

**INSERT** **INTO** empleados (nombre, edad, puesto, salario)

**VALUES**

('María López', 28, 'Analista', 45000),

('Carlos Gómez', 35, 'Gerente', 70000);

📌 **Ejemplo: Insertar datos sin especificar algunas columnas (tomará valores por defecto o NULL)**

**INSERT** **INTO** empleados (nombre, edad)

**VALUES** ('Ana Ramírez', 27);

Como se ve en los ejemplos anteriores los Campos ID (identity) no se especifican en la inserción de los datos ya que son campos auto-incrementales, su valor se va a insertar automáticamente cuando se ingrese un registro nuevo a la tabla

**UPDATE – Modificar registros existentes**

Se usa para actualizar valores en una o varias filas de una tabla.

📌 **Ejemplo: Actualizar el salario de un empleado específico**

**UPDATE** empleados

**SET** salario = 55000

**WHERE** id = 1;

📌 **Ejemplo: Aumentar un 10% el salario de todos los empleados**

**UPDATE** empleados

**SET** salario = salario \* 1.10;

📌 **Ejemplo: Cambiar el puesto de los empleados con un salario menor a 50000**

**UPDATE** empleados

**SET** puesto = 'Junior'

**WHERE** salario < 50000;

**DELETE – Eliminar registros de una tabla**

Borra filas específicas según una condición.

📌 **Ejemplo: Eliminar un solo registro**

**DELETE** **FROM** empleados

**WHERE** id = 3;

📌 **Ejemplo: Eliminar todos los empleados con salario menor a 30000**

**DELETE** **FROM** empleados

**WHERE** salario < 30000;

⚠ **¡Cuidado! Si no pones WHERE, borras todos los datos de la tabla.**

📌 **Ejemplo: Borrar todos los registros de la tabla (similar a TRUNCATE)**

**DELETE** **FROM** empleados;

**TRUNCATE – Eliminar registros de una tabla rápidamente**

Al igual que el DELETE elimina todos los registros de una tabla. Es más eficiente que DELETE porque no genera registros en el log de transacciones. Pero a diferencia del DELETE **en** **el truncate no se puede usar la sentencia WHERE**, por lo que si se desea borrar algunos registros solamente deberemos usar la sentencia DELETE. Otra diferencia entre estas 2 sentencias es que el TRUNCATE reinicia los ID automáticos (IDENTITY) por lo que si la tabla tenia por ejemplo 10 registros al hacer un truncate el campo ID de la tabla volverá a 1.

**Cuando usar TRUNCATE en lugar de DELETE**

* Si necesitas **borrar todos los datos** de una tabla rápidamente.
* Si no quieres afectar el log de transacciones.
* Si quieres **resetear los valores de IDENTITY** en columnas autoincreméntales.

📌 **Ejemplo: Vaciar una tabla**

**TRUNCATE TABLE** empleados;

**📝 Actividad**

Para poder realizar las actividades vamos a necesitar crear previamente la siguiente tabla:

Clientes

* id (INT, clave primaria) **IDENTITY(1,1) PRIMARY KEY NOT NULL**
* nombre (VARCHAR(255))
* correo (VARCHAR(255))
* edad (INT)
* saldo (DECIMAL(10,2))

1. Inserta 5 registros en la tabla clientes.
2. Modifica la edad de un cliente específico (referenciar por id).
3. Aumenta un 5% el saldo de todos los clientes.
4. Borrar los clientes que no tengan correo.
5. Truncar la tabla clientes.