**¿Cuál es la diferencia entre las acciones ON DELETE y ON UPDATE en una llave foránea al usar las opciones CASCADE, SET NULL y RESTRICT, ¿y cómo se implementan correctamente en una consulta SQL?**

**CASCADE**

Si se elimina o actualiza un registro en la tabla padre, los registros relacionados en la tabla hija también se eliminan o actualizan automáticamente.

Ejemplo típico: si borras un estudiante, también se eliminan sus calificaciones.

**SET NULL**

Si se elimina o actualiza un registro en la tabla padre, la columna de la llave foránea en la tabla hija se establece en NULL.

Necesitas que la columna tenga permitido valores nulos.

**RESTRICT (o su equivalente NO ACTION en algunos motores)**

No permite eliminar o actualizar un registro de la tabla padre si existen registros relacionados en la tabla hija.

Es una forma de proteger la integridad referencial.

**¿Qué es la restricción CHECK?**

La restricción CHECK se usa para limitar los valores que se pueden insertar en una columna, según una condición lógica.

Es como decirle a la base de datos: "solo acepta esto si cumple tal regla".

**1. INSERT INTO ... VALUES**

Se usa para insertar datos manualmente, fila por fila.

INSERT INTO Estudiantes (id\_estudiante, nombre, edad)

VALUES (1, 'Camila Torres', 20);

**2. INSERT INTO ... SELECT**

Se usa para copiar datos desde otra tabla (o resultado de una consulta) hacia una tabla destino.

-- Supongamos que ya tienes la tabla Estudiantes

-- Y ahora quieres crear una tabla llamada EstudiantesBackup con los mismos datos:

INSERT INTO EstudiantesBackup (id\_estudiante, nombre, edad)

SELECT id\_estudiante, nombre, edad FROM Estudiantes;

**3. SELECT ... INTO (a veces llamado "select insert")**

Crea una nueva tabla desde cero y le mete datos en una sola línea.

SELECT id\_estudiante, nombre, edad

INTO EstudiantesCopia

FROM Estudiantes

WHERE edad > 18;

Crea la tabla EstudiantesCopia automáticamente.

Le inserta los estudiantes mayores de 18 años.

Esto solo funciona si la tabla destino no existe aún.

**1. UPDATE con WHERE**

Se usa cuando quieres modificar solo ciertos registros que cumplan una condición.

-- Cambiar el nombre de la asignatura con ID 2

UPDATE Asignaturas

SET nombre = 'Frontend Intermedio'

WHERE id\_asignatura = 2;

**2. UPDATE sin WHERE**

Se aplica a TODAS las filas de la tabla.

-- Cambiar el nombre de TODAS las asignaturas a "Asignatura temporal"

UPDATE Asignaturas

SET nombre = 'Asignatura temporal';

Esto cambia todas las filas.

**Inyección SQL:**

La inyección SQL (SQL Injection) es culpa del programador, no del motor de base de datos.

Porque la inyección SQL solo ocurre si el código de la aplicación está mal hecho, es decir, si:

* El programador no valida la entrada del usuario
* Construye las consultas SQL como cadenas de texto sin protección
* No usa consultas preparadas o parámetros

**¿Cómo evitarlo? (Solución del programador)**

* Usar consultas preparadas (prepared statements)
* Validar/sanitizar las entradas del usuario
* Limitar los privilegios del usuario de la base de datos

**¿Qué son las consultas principales?**

Son las consultas normales de toda la vida.

**¿Qué son las subconsultas?**

Son consultas anidadas dentro de otra consulta.

Tipos de subconsultas:

* En el WHERE
* En el FROM
* En el SELECT

1. Subconsulta en el WHERE (la más común)

SELECT nombre

FROM Estudiantes

WHERE id\_estudiante IN (

SELECT id\_estudiante

FROM Notas

WHERE calificacion < 3.0

);

**2. Subconsulta en el FROM (subtabla temporal)**

SELECT promedio\_general FROM (

SELECT AVG(calificacion) AS promedio\_general FROM Notas) AS promedio;

**3. Subconsulta en el SELECT**

SELECT nombre,

(SELECT COUNT(\*) FROM Notas WHERE Notas.id\_estudiante = Estudiantes.id\_estudiante) AS total\_notas

FROM Estudiantes;