¡Sí, en general, tienes bien los conceptos! Te hago algunas correcciones y aclaraciones para que lo tengas claro:

✅ **INSERT** → Se usa para insertar registros en una tabla.

INSERT INTO tabla (columna1, columna2) VALUES ('valor1', 'valor2');

✅ **UPDATE** → Se usa para modificar registros existentes en una tabla. Debes usar WHERE para evitar cambiar todos los registros.

UPDATE tabla SET columna1 = 'nuevo\_valor' WHERE columna2 = 'condicion';

✅ **DELETE** → Se usa para borrar registros específicos de una tabla. Si no usas WHERE, borra **todos** los registros.

DELETE FROM tabla WHERE columna1 = 'valor';

✅ **ALTER TABLE ADD** → Se usa para añadir columnas o restricciones a una tabla.

ALTER TABLE tabla ADD nueva\_columna VARCHAR2(50);

ALTER TABLE tabla ADD CONSTRAINT pk\_tabla PRIMARY KEY (columna1);

⚠️ **MODIFY** → No se usa para cambiar el nombre de una columna, sino para modificar su tipo de dato o restricciones.

ALTER TABLE tabla MODIFY columna1 NUMBER(10);

⚠️ **RENAME COLUMN** → Para cambiar el nombre de una columna en Oracle, usas RENAME COLUMN:

ALTER TABLE tabla RENAME COLUMN columna\_vieja TO columna\_nueva;

✅ **DROP** → Se usa para eliminar columnas, restricciones o tablas enteras.

* Borrar una columna:
* ALTER TABLE tabla DROP COLUMN columna1;
* Borrar una restricción:
* ALTER TABLE tabla DROP CONSTRAINT nombre\_restriccion;
* Borrar una tabla entera:
* DROP TABLE tabla;

Si necesitas más detalles o ejemplos, dime qué parte quieres reforzar. 🔥

En **Oracle SQL**, tanto TRUNCATE como DROP se utilizan para eliminar datos de una tabla, pero tienen diferencias clave en su funcionamiento:

**1. TRUNCATE**

* **Elimina** todos los registros de una tabla sin eliminar la estructura de la tabla.
* Es más rápido que DELETE porque no genera registros en el log de transacciones de la misma manera.
* No se pueden hacer **ROLLBACK** después de ejecutar TRUNCATE, ya que es una operación **DDL** (Data Definition Language).
* **No activa triggers** porque no opera fila por fila.
* **Mantiene la estructura** de la tabla para poder seguir insertando datos.

**Ejemplo:**

TRUNCATE TABLE empleados;

**2. DROP**

* **Elimina** completamente la tabla de la base de datos, incluyendo su estructura, índices y permisos.
* No se puede recuperar la tabla a menos que se use FLASHBACK TABLE (si está habilitado).
* Es una operación **DDL**, por lo que no se puede hacer **ROLLBACK**.
* **También puede eliminar otros objetos**, como vistas, índices y procedimientos.

**Ejemplo:**

DROP TABLE empleados;

**Diferencias Claves**

| **Característica** | **TRUNCATE** | **DROP** |
| --- | --- | --- |
| Elimina datos | ✅ Sí | ✅ Sí |
| Elimina la estructura | ❌ No | ✅ Sí |
| Se puede hacer ROLLBACK | ❌ No | ❌ No |
| Es rápido | ✅ Sí | ✅ Sí |
| Afecta a índices | ✅ (Reinicia los índices) | ✅ (Elimina los índices) |
| Mantiene permisos y estructura | ✅ Sí | ❌ No |

En resumen:

* Usa TRUNCATE cuando solo necesitas borrar los datos pero mantener la tabla.
* Usa DROP cuando quieres eliminar por completo la tabla y su estructura.

✅ **SAVEPOINT** en Oracle SQL es un comando que permite establecer un "punto de guardado" dentro de una transacción. Esto permite hacer un ROLLBACK parcial, es decir, deshacer solo los cambios desde un punto específico sin afectar toda la transacción.

**Ejemplo de Uso**

-- Iniciar una transacción

INSERT INTO empleados (id, nombre, salario) VALUES (1, 'Juan', 3000);

SAVEPOINT sp1; -- Guardamos este punto

INSERT INTO empleados (id, nombre, salario) VALUES (2, 'Ana', 3500);

SAVEPOINT sp2; -- Otro punto de guardado

UPDATE empleados SET salario = 4000 WHERE id = 1;

🔹 **Si hacemos ROLLBACK TO sp2**, se deshace el UPDATE, pero la inserción de "Ana" sigue existiendo:

ROLLBACK TO sp2;

🔹 **Si hacemos ROLLBACK TO sp1**, se deshace todo lo que ocurrió después de sp1, eliminando tanto la inserción de "Ana" como el UPDATE:

ROLLBACK TO sp1;

🔹 **Si hacemos un ROLLBACK sin TO**, se deshace todo lo que no se haya confirmado con COMMIT:

ROLLBACK;

📌 **Notas Clave**:

* Los SAVEPOINT permiten hacer reversión **parcial** dentro de una transacción.
* Si haces COMMIT, todos los SAVEPOINT se eliminan, ya que los cambios quedan guardados de forma permanente.
* Si haces un ROLLBACK sin indicar un SAVEPOINT, se desharán **todos** los cambios de la transacción.

¿Necesitas más ejemplos o alguna aclaración? 🚀