**Teoría de Transacciones en PL/SQL**

Una **transacción** es una unidad lógica de trabajo que agrupa una o más operaciones SQL (INSERT, UPDATE, DELETE). En PL/SQL, una transacción sigue el principio **ACID**, que garantiza que los datos sean consistentes y fiables.

**Principios ACID de una Transacción**

1. **Atomicidad (Atomicity)**: La transacción se ejecuta completamente o no se ejecuta nada.
2. **Consistencia (Consistency)**: Los datos pasan de un estado válido a otro, manteniendo la integridad.
3. **Aislamiento (Isolation)**: Una transacción no afecta ni es afectada por otras transacciones concurrentes.
4. **Durabilidad (Durability)**: Una vez confirmada, la transacción queda guardada permanentemente.

**Control de Transacciones en PL/SQL**

**1️⃣ COMMIT**

Guarda los cambios realizados por la transacción en la base de datos. Una vez ejecutado COMMIT, los cambios se vuelven permanentes y visibles para otros usuarios.

UPDATE EMP SET SAL = SAL \* 1.10 WHERE JOB = 'CLERK';

COMMIT; -- Se guardan los cambios

**2️⃣ ROLLBACK**

Revierte todos los cambios no confirmados desde el último COMMIT. Se usa cuando hay un error y queremos deshacer las modificaciones.

UPDATE EMP SET SAL = SAL \* 1.10 WHERE JOB = 'CLERK';

ROLLBACK; -- Deshace los cambios y vuelve al estado original

**3️⃣ SAVEPOINT**

Permite definir puntos dentro de una transacción. Si ocurre un error, podemos hacer un ROLLBACK TO savepoint\_name en lugar de deshacer toda la transacción.

SAVEPOINT punto1;

UPDATE EMP SET SAL = SAL \* 1.10 WHERE JOB = 'CLERK';

SAVEPOINT punto2;

UPDATE EMP SET SAL = SAL \* 1.20 WHERE JOB = 'MANAGER';

ROLLBACK TO punto1; -- Solo deshace la actualización de MANAGER, pero mantiene la de CLERK

COMMIT; -- Guarda los cambios de CLERK

**4️⃣ Manejo de Errores en una Transacción**

Si hay un error, podemos capturarlo con EXCEPTION y hacer un ROLLBACK.

BEGIN

UPDATE EMP SET SAL = SAL \* 1.10 WHERE JOB = 'CLERK';

COMMIT;

EXCEPTION

WHEN OTHERS THEN

ROLLBACK;

DBMS\_OUTPUT.PUT\_LINE('Error: ' || SQLERRM);

END;

**Resumen**

| **Comando** | **Descripción** |
| --- | --- |
| COMMIT | Guarda los cambios en la base de datos. |
| ROLLBACK | Revierte todos los cambios no guardados. |
| SAVEPOINT | Crea un punto de control en la transacción. |
| ROLLBACK TO savepoint\_name | Revierte solo hasta un punto específico. |

✅ **Conclusión**: Las transacciones aseguran que los datos sean confiables y evitan inconsistencias si ocurre un error.