## .Manipulación de datos

El lenguaje de manipulación de datos o DML (*Data Manipulation Language*) es una parte esencial de SQL. Cuando se quiere agregar, actualizar o eliminar datos, se ejecuta una sentencia DML. Un conjunto de sentencias DML que aún no se han hecho permanentes se denomina **transacción** o unidad lógica de trabajo.

Imagine una base de datos de un banco. Cuando un cliente del banco transfiere dinero de una cuenta a otra, la transacción consiste en tres operaciones separadas: disminuir el saldo, incrementar el de la otra cuenta y registrar la transacción. Oracle debe garantizar que las tres sentencias SQL se ejecutan para mantener las cuentas con el saldo adecuado. Cuando una de las sentencias no puede ejecutarse, el resto de las sentencias deben deshacerse.

## La sentencia INSERT

Permite añadir nuevos registros a una tabla.

Sintaxis:

INSERT INTO tabla [columna [, columna ...])]

VALUES (valor [, valor ...]);

Mediante esta sintaxis, sólo se inserta un registro o fila al mismo tiempo.

Ejemplo:

INSERT INTO DEPT (deptno, dname, loc)

VALUES (50, ‘DESARROLLO’, ‘MADRID’);

Es preciso listar los valores en el orden por defecto de las columnas de la tabla.

Opcionalmente se pueden listar las columnas en las cláusula INSERT.

Si se desean insertar valores nulos, se pueden utilizar dos métodos:

* Método implícito: omitir la columna en la lista.

INSERT INTO DEPT (deptno, dname)

VALUES (60, ‘MI DEPARTAMENTO’);

* Método explícito: especificar la palabra clave NULL.

INSERT INTO DEPT

VALUES (60, ‘MI DEPARTAMENTO’, NULL);

Antes hay que asegurarse que la columna permite valores nulos (comprobarlo con DESCRIBE).

Ejemplos:

INSERT INTO EMP (empno, ename, job, mgr, hiredate, sal, comm, deptno)

VALUES (7196, ‘HIGUERAS’, ‘SALESMAN’, 7782, SYSDATE, 2000, NULL, 10);

INSERT INTO EMP

VALUES (2296, ‘ORTEGA’, ‘SALESMAN’, 7782,

TO\_DATE (‘FEB 3, 97’, ‘MON DD, YY’), 1300, NULL, 10);

**Copia de registros de otra tabla**

Se pueden usar la sentencia INSERT para agregar filas a una tabla donde los valores se derivan de otras tablas ya existentes. En lugar de la cláusula VALUES, utilizamos una subconsulta.

INSERT INTO JEFES (id, nombre, salario, alta)

SELECT empno, ename, sal, hiredate FROM EMP WHERE job = ‘MANAGER’;

Cuidar la concordancia de columnas de la cláusula INSERT y las de la subconsulta.

## La sentencia UPDATE

Permite modificar los registros existentes. Puede afectar a más de un registro al mismo tiempo.

Sintaxis:

UPDATE tabla

SET columna = valor [, columna = valor]

[WHERE condición];

Los registros a modificar se especifican en la cláusula WHERE.

UPDATE EMP

SET deptno = 20

WHERE empno = 7782;

Si se omite la cláusula WHERE, se modificarán **todos** los registros de la tabla.

UPDATE EMPLEADOS

SET deptno = 20;

Ejemplos:

UPDATE EMP

SET (job, deptno) = (SELECT job, deptno

FROM EMP

WHERE empno =7499)

WHERE empno = 7698;

UPDATE EMP

SET deptno = 55

WHERE deptno = 10;

SQL>> ERROR: violación de restricción de integridad (el departamento 55 no existe en la tabla padre DEPT).

## La sentencia DELETE

Permite borrar registros de una tabla.

Sintaxis:

DELETE [FROM] tabla

[WHERE condición];

Los registros a eliminar se especifican en la cláusula WHERE.

DELETE FROM DEPARTAMENTOS

WHERE dname = ‘DEVELOPMENT’;

Si se omite la cláusula WHERE se borrarán **todos** los registros de la tabla.

DELETE FROM DEPARTAMENTOS;

Ejemplos:

DELETE FROM EMPLEADOS

WHERE deptno = (SELECT deptno

FROM DEPT

WHERE dname = ‘SALES’);

DELETE FROM DEPT

WHERE deptno = 10;

SQL>> ERROR: violación de restricción de integridad (no se puede eliminar un registro que contiene una clave primaria, usada como clave extranjera en otra tabla).

## 7.Transacciones en la base de datos

El servidor Oracle asegura la consistencia de los datos basándose en transacciones. Las transacciones otorgan más flexibilidad y control cuando cambian los datos y aseguran en los datos ante un fallo eventual en el proceso del usuario o un fallo del sistema.

Las transacciones consisten de comandos DML que llevan a cabo un cambio en los datos de manera consistente. Por ejemplo, una transferencia de fondos entre dos cuentas, debiera incluir un crédito en una cuenta y un débito en la otra por la misma cantidad. Ambas acciones debieran tener éxito o fallar como una sola operación. El crédito no debería confirmarse sin el débito correspondientes.

Tipos de transacciones:

* Un conjunto de sentencias DML que Oracle tratará como una unidad simple
* Una única sentencia DDL
* Una única sentencia DCL

Las transacciones comienzan cuando se ejecuta el primer comando SQL.

Las transacciones finalizan con uno de los siguientes eventos:

* COMMIT o ROLLBACK
* Ejecución de un comando DCL o DCL (COMMIT automático)
* Salida del usuario
* Caída del sistema

Las sentencias COMMIT y ROLLBACK aseguran la consistencia de los datos, permiten visualizar los cambios sobre los datos antes de hacerlos permanentes y agrupan lógicamente tareas relacionadas entre sí.