

LANGAGE DE MANIPULATION DES DONNEES LMD

LANGAGE DE MANIPULATION DES DONNEES LMD

1. INSERT

2. UPDATE

3. DELETE

LANGAGE DE MANIPULATION DES DONNEES LMD

Un ordre du langage LMD est exécuté lorsque :

- **Vous ajoutez des lignes à une table**
- **Vous modifiez des lignes existantes dans une table**
- **Vous supprimez des lignes dans une table**

Une transaction est un ensemble d'ordres du LMD formant une unité de travail logique

1.INSERT

1.INSERT INTO TABLE VALUES

(value1, value2, ... valuen)

Exemple

INSERT INTO dept VALUES (50, 'DEVELOPMENT', WASHINGTON')

INSERTION DE LIGNES EN PRECISANT LES COLONNES

- 1.INSERT INTO TABLE(col1, col2, ... coln) VALUES(val1, val2, ... valn)**
- 2.L'ordre des colonnes et des valeurs doivent être le même**
- 3.Placez les valeurs de type caractère et date entre simples quotes**

Exemples

INSERT INTO dept(deptno,dname,loc) VALUES (50,'DEVELOPMENT', WASHINGTON')

INSERT INTO dept(loc,dname,deptno) VALUES ('WASHINGTON', 'DEVELOPMENT', 50)

INSERTION DES LIGNES CONTENANT DES VALEURS NULL

1. Méthode explicite

INSERT INTO dept(deptno,dname) VALUES(50,'DEVELOPMENT')

2. Méthode implicite

3. INSERT INTO dept VALUES(50,'DEVELOPMENT', NULL)

INSERTION DE VALEURS SPECIFIQUES

1. SYSDATE

```
INSERT INTO EMP(empno, ename, job, mgr ,hiredate, sal, comm, deptno)  
VALUES (7196, 'GREEN','SALESMAN', 7782, SYSDATE, 2000, NULL, 10)
```

2. INSERTION DE DATE DANS UN FORMAT SPECIFIQUE

```
INSERT INTO EMP(empno, ename, job, mgr, hiredate, sal, comm, deptno) VALUES  
(2296,'AROMANO','SALESMAN', 7782, TO_DATE('3 Févr. 2007', 'DD,MON,YYYY'), 1300, NULL, 10)
```

INSERTION A PARTIR D'UNE AUTRE TABLE

1. **INSERT INTO TABLE1(col1, col2, ... coln)**
SELECT col1, col2, ..., coln
FROM Table2
2. Le nombre de colonnes de la clause INSERT doit correspondre à celui de la sous-interrogation

Exemple

```
INSERT INTO managers(id, name, salary, hiredate)  
SELECT empno, ename, sal, hiredate  
FROM emp WHERE job='MANAGER'
```


2.UPDATE

L'ordre update permet de modifier des lignes d'une table

UPDATE Table SET col1=val1 [,col2 = val2]... [,coln = valn]

[WHERE conditions]

Exemple

UPDATE emp SET deptno=20 WHERE empno=7782

(le département de l'employé 7782 est mis à jour)

UPDATE emp SET deptno=20

(le département de tous les employés est mis à jour)

MODIFICATION AVEC UNE SOUS-INTERROGATION MULTI-COLONNES

```
UPDATE Table 1 SET (col1, col2,...coln) =  
(SELECT col1, col2, ...coln FROM Table2 WHERE condition )  
WHERE condition
```

Exemple

```
UPDATE emp SET (job, deptno)=  
(SELECT job, deptno FROM emp WHERE empno=7499)  
WHERE empno=7698
```

MODIFICATION DE LIGNES EN FONCTION D'UNE AUTRE TABLE

UPDATE Table1 as a

SET coli= (SELECT expression FROM Table2 b

WHERE a.col1=b.col2

Exemple

ALTER TABLE emp ADD (dname varchar2(14))

UPDATE emp e

SET dname = (SELECT dname FROM dept d WHERE e.deptno=d.deptno)

DELETE

DELETE [FROM] Table WHERE condition

Exemple

DELETE FROM dept WHERE dname='DEVELOPMENT'

(le département DEVELOPMENT est supprimé)

DELETE FROM emp

WHERE hiredate > to_date('01.01.97', 'DD.MM.YY')

(tous les employés recrutés après le 01.01.97 sont supprimés)

DELETE FROM dept

(tous les départements sont supprimés)

SUPPRESSION DE LIGNES EN FAISANT REFERENCE A UNE AUTRE TABLE

DELETE FROM EMP

WHERE Deptno=

(SELECT deptno FROM dept WHERE dname='SALES')

SUPPRESSION SYNCHRONISEE

```
DELETE FROM Table1 a  
WHERE colonne operator (SELECT expression FROM Table2 b  
WHERE a.colonne1=b.colonne2)
```