

Limbajul de manipulare a datelor (LMD)

- Comenzile SQL care alcătuiesc **LMD** permit:
 - regăsirea datelor (*SELECT*);
 - adăugarea de noi înregistrări (*INSERT*);
 - modificarea valorilor coloanelor din înregistrările existente (*UPDATE*);
 - adăugarea sau modificarea condiționată de înregistrări (*MERGE*);
 - suprimarea de înregistrări (*DELETE*).

I. Comanda *INSERT*

1. Inserări mono-tabel

Comanda *INSERT* are următoarea sintaxă simplificată:

```
INSERT INTO obiect [AS alias] [ (nume_coloană [, nume_coloană ...] ) ]  
{ VALUES ( {expr | DEFAULT} [, {expr | DEFAULT} ...] )  
  | subcerere }
```

Subcererea specificată în comanda *INSERT* returnează linii care vor fi adăugate în tabel.

Dacă în tabel se introduc linii prin intermediul unei subcereri, coloanele din lista *SELECT* trebuie să corespundă, ca număr și tip, celor precizate în clauza *INTO*. În absența unei liste de coloane în clauza *INTO*, subcererea trebuie să furnizeze valori pentru fiecare atribut al obiectului destinație, respectând ordinea în care acestea au fost definite.

Observații (tipuri de date):

- Pentru claritate, este recomandată utilizarea unei liste de coloane în clauza *INSERT*.
- În clauza *VALUES*, valorile de tip caracter și dată calendaristică trebuie incluse între apostrofuri. Nu se recomandă includerea între apostrofuri a valorilor numerice, întrucât aceasta ar determina conversii implicite la tipul *NUMBER*.
- Pentru introducerea de valori speciale în tabel, pot fi utilizate funcții.

Adăugarea unei linii care va conține valori *null* se poate realiza în mod:

- implicit, prin omiterea numelui coloanei din lista de coloane;
- explicit, prin specificarea în lista de valori a cuvântului cheie *null*

În cazul șirurilor de caractere sau al datelor calendaristice se poate preciza șirul vid ("").

Observații (erori):

Server-ul *Oracle* aplică automat toate tipurile de date, domeniile de valori și constrângerile de integritate. La introducerea sau actualizarea de înregistrări, pot apărea erori în următoarele situații:

- nu a fost specificată o valoare pentru o coloană *NOT NULL*;
- există valori duplicate care încalcă o constrângere de unicitate;
- a fost încălcată constrângerea de cheie externă sau o constrângere de tip *CHECK*;
- există o incompatibilitate în privința tipurilor de date;
- s-a încercat inserarea unei valori având o dimensiune mai mare decât a coloanei corespunzătoare.

2. Inserari multi-tabel

O inserare multi-tabel presupune introducerea de linii calculate pe baza rezultatelor unei subcereri, într-unul sau mai multe tabele. Acest tip de inserare, introdus de *Oracle9i*, este util în mediul *data warehouse*.

Pentru o astfel de inserare, în versiunile anterioare lui *Oracle9i* erau necesare n operații independente *INSERT INTO...SELECT...*, unde n reprezintă numărul tabelelor destinație. Aceasta presupunea n procesări ale aceleiași surse de date și, prin urmare, creșterea de n ori a timpului necesar procesului.

Sintaxa comenzii *INSERT* în acest caz poate fi:

- Pentru inserări necondiționate:

```
INSERT ALL INTO... [INTO...]  
subcerere;
```

- Pentru inserări condiționate:

```
INSERT [ALL | FIRST]  
WHEN condiție THEN INTO...  
[WHEN condiție THEN INTO...  
[ELSE INTO ...]]  
subcerere;
```

- *ALL* determină evaluarea tuturor condițiilor din clauzele *WHEN*. Pentru cele a căror valoare este *TRUE*, se inserează înregistrarea specificată în opțiunea *INTO* corespunzătoare.

- *FIRST* determină inserarea corespunzătoare primei clauze *WHEN* a cărei condiție este evaluată *TRUE*. Toate celelalte clauze *WHEN* sunt ignorate.

II. Comanda **UPDATE**

Sintaxa simplificată a comenzii *UPDATE* este:

```
UPDATE nume_tabel [alias]  
SET col1 = expr1[, col2=expr2]  
[WHERE condiție];
```

sau

```
UPDATE nume_tabel [alias]  
SET (col1,col2,...) = (subcerere)  
[WHERE condiție];
```

Observații:

- de obicei pentru identificarea unei linii se folosește o condiție ce implică cheia primară;
- dacă nu apare clauza *WHERE* atunci sunt afectate toate liniile tabelului specificat;
- cazurile în care instrucțiunea *UPDATE* nu poate fi executată sunt similare celor în care eșuează instrucțiunea *INSERT*. Acestea au fost menționate anterior.

III. Comanda **DELETE**

Sintaxa simplificată a comenzii *DELETE* este:

```
DELETE FROM nume_tabel  
[WHERE condiție];
```

Dacă nu se specifica nici o condiție, vor fi șterse toate liniile din tabel.

Exerciții [I]

1. Să se insereze departamentul 300, cu numele *Programare* în *DEPT_pnu* (copie a tabelului *Departments*).

Analizați cazurile, precizând care este soluția corectă și explicând erorile celorlalte variante. Pentru a anula efectul instrucțiunii(ilor) corecte, utilizați comanda *ROLLBACK*.

- a) *INSERT INTO DEPT_pnu*
VALUES (300, 'Programare');
- b) *INSERT INTO DEPT_pnu (department_id, department_name)*
VALUES (300, 'Programare');
- c) *INSERT INTO DEPT_pnu (department_name, department_id)*
VALUES (300, 'Programare');
- d) *INSERT INTO DEPT_pnu (department_id, department_name, location_id)*
VALUES (300, 'Programare', null);
- e) *INSERT INTO DEPT_pnu (department_name, location_id)*
VALUES ('Programare', null);

2. Încercați dacă este posibilă introducerea unui angajat, precizând pentru valoarea *employee_id* o subcerere care returnează (codul maxim +1).
3. Creați un nou tabel, numit *EMP1_PNU*, care va avea aceeași structură ca și *EMPLOYEES*, dar nicio înregistrare. Copiați în tabelul *EMP1_PNU* salariații (din tabelul *EMPLOYEES*) al căror salariu este cel puțin 10000.
4. Să se creeze tabelele necesare cu aceeași structură ca a tabelului *EMPLOYEES* (fără constrângeri și fără înregistrări). Copiați din tabelul *EMPLOYEES*:
 - în tabelul *EMP0_PNU* salariații care lucrează în departamentul 80;
 - în tabelul *EMP1_PNU* salariații care au salariul mai mic decât 5000;
 - în tabelul *EMP2_PNU* salariații care au salariul cuprins între 5000 și 10000;
 - în tabelul *EMP3_PNU* salariații care au salariul mai mare decât 10000.

Dacă un salariat se încadrează în tabelul *emp0_pnu* atunci acesta nu va mai fi inserat și în alt tabel (tabelul corespunzător salariului său).

Exerciții [II]

5. Măriți salariul tuturor angajaților din tabelul *EMP_PNU* cu 5%. Vizualizați, iar apoi anulați modificările.
6. Să se promoveze Douglas Grant la manager în departamentul 20, având o creștere de salariu cu 1000\$. Se poate realiza modificarea prin intermediul unei singure comenzi?
7. Să se modifice jobul și departamentul angajatului având codul 114, astfel încât să fie la fel cu cele ale angajatului având codul 205.

Exerciții [III]

8. Ștergeți toate înregistrările din tabelul *DEPT_PNU*. Ce înregistrări se pot șterge? Anulați modificările.
9. Ștergeți angajații care nu au comision. Anulați modificările.

