**Laborator 7**

1. Inserarea unui angajat in tabel.

insert into employees\_eis

values (252, null, 'Enescu', 'irina@gmail.com', null, sysdate, 'test\_job', null, null, null, 300);

Alta varianta:

insert into employees\_eis (employee\_id, last\_name, email, hire\_date, job\_id, department\_id)

values (253, 'Enescu', 'irina@gmail.com', sysdate, 'test\_job', 300);

Atentie!! – EROARE ORA-02291: integrity constraint (GRUPA33.FK\_EMPLOYEES\_DEPARTMENTS\_EIS) violated - parent key not found (nu a fost creat departamentul)

insert into employees\_eis (employee\_id, last\_name, email, hire\_date, job\_id, department\_id)

values (254, 'Enescu', 'irina@gmail.com', sysdate, 'test\_job', 301);

1. Copiati structura unui tabel.

create table emp1\_eis as select \* from employees where 1 = 0;

(deoarece conditia este imposibila, nu se vor copia si date, ci doar structura)

Alta varianta:

create table emp1\_eis as select \* from employees;

delete from emp1\_eis;

(intai se va copia tabelul cu date si apoi se vor sterge toate datele din el)

ATENTIE!! NU SE SETEAZA CHEIA PRIMARA >>> TREBUIE SA O SETAM NOI AUTOMAT. De aceea se pot introduce la exercitiul 4 de cate ori vrem noi angajati cu acelasi employee\_id.

1. Stergerea unui tabel.

drop table emp1\_eis;

1. Inserare in tabel angajati cu comision mai are decat 25%.

insert into emp1\_eis

select \* from employees where commission\_pct > 0.25;

1. Comanda de UPDATE – modificare salariu.

UPDATE pentru toti angajatii.

update employees\_eis set salary = 10000;

UPDATE pentru angajatii introdusi la exercitiile anterioare.

update employees\_eis set salary = 700

where lower(job\_id) = 'test\_job';

Daca dam rollback se revine la ce aveam inainte!! Putem face si asa:

update employees\_eis e

set e.salary = (select a.salary

from employees a

where a.employee\_id = e.employee\_id);

Dar la angajatii introdusi de noi avem null deoarece nu exista in celalat tabel.

1. Marire salariu cu 5% pentru toti angajatii.

Este indicat sa se foloseasca functia NVL(salary + … , 0) !!

update employees\_eis set salary = salary + salary\*0.05;

1. Schimbare job in SA\_REP pentru angajatii din departamentul 80 cu commission.

update employees\_eis set job\_id = 'SA\_REP'

where department\_id = 80 and commission\_pct is not null;

1. Sa se promoveze Douglas Grant la functia de manager in departamentul 20, avand o crestere de salariu de 1000.

-- UPDATE MANAGER

update departments\_eis set manager\_id =

(select employee\_id from employees\_eis where lower(first\_name) = 'douglas' and lower(last\_name) = 'grant')

where department\_id = 20;

-- UPDATE DEPARTAMENT IN CARE LUCREAZA si SALARIU

update employees\_eis set department\_id = 20, salary = salary + 1000

where lower(first\_name) = 'douglas' and lower(last\_name) = 'grant';

1. Stergerea departamentelor in care nu sunt angajati.

delete from departments\_eis

where department\_id in

(select department\_id

from departments\_eis

minus

select unique department\_id

from employees\_eis);