ALTER TABLE emp\_bai //un dep are 1/M employees

ADD CONSTRAINT fk\_emp\_dept\_bai //emp primeste FK dep\_id

FOREIGN KEY(department\_id) REFERENCES dept\_bai(department\_id);

ALTER TABLE dept\_bai

ADD CONSTRAINT fk\_dept\_emp\_bai

FOREIGN KEY(manager\_id) REFERENCES emp\_bai(employee\_id) disable novalidate;

-- nu doresc sa fie o regula pe care sa o faca datele, de aceea e disable

Insert into dept\_bai values (200, 'programare'); -- "not enough values"

insert into dept\_bai (department\_id, department\_name)

--specify the exact columns ' values to insert into

values (300, 'programare');

INSERT INTO DEPT\_bai (department\_id, department\_name, location\_id)

VALUES (300, 'Programare fara locatie', null);

!!!!! TOATE COLOANELE MENTIONATE

INSERT INTO DEPT\_pnu (department\_name, location\_id)

VALUES (‘Programare’, null); -- does not work since pk is not autoincrementable

-- Să se mai introducă un angajat corespunzător departamentului 300, precizând după numele tabelului lista coloanelor în care se introduc valori (metoda explicita de inserare). Se presupune că data angajării acestuia este cea curentă (SYSDATE). Salvaţi înregistrarea.

insert into emp\_bai (employee\_id, last\_name, first\_name, email, hire\_date, job\_id, department\_id)

values( 1001, 'Vioreanu','Florin', 'fc@yahoo.com', sysdate, 'It\_prog', 300);

commit;

--7. Este posibilă introducerea de înregistrări prin intermediul subcererilor (specificate în locul tabelului). Ce reprezintă, de fapt, aceste subcereri? (view) Încercaţi dacă este posibilă introducerea unui angajat, precizând pentru valoarea employee\_id o subcerere care returnează (codul maxim +1)

insert into emp\_bai (employee\_id, last\_name, first\_name, email, hire\_date, job\_id, department\_id)

select max (employee\_id) +1 , 'Vioreanu','Florina', 'fc@yahoo.com', sysdate, 'It\_prog', 300;

from emp\_bai;

NU MAI APARE VALUES

rollback; // anuleaza toate modif facute incepand de la ultimul commit -- undo

commit;

--8. Creaţi un nou tabel, numit EMP1\_PNU, care va avea aceeaşi structură ca şi EMPLOYEES, dar nici o înregistrare. Copiaţi în tabelul EMP1\_PNU salariaţii (din tabelul EMPLOYEES) al căror comision depăşeşte 25% din salariu.

create table emp\_bai2

as select \* from employees where 1 = 2;//pt a nu avea nicio linie, ci doar FORMA

insert into emp\_bai2

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

-- 10. Măriţi salariul tuturor angajaţilor din tabelul EMP\_PNU cu 5%. Vizualizati, iar apoi anulaţi modificările.

update emp\_bai2 set salary = salary \* 1.05;

--15. Schimbaţi jobul tuturor salariaţilor din departamentul 80 care au comision în 'SA\_REP'. Anulaţi modificările

update emp\_bai

set job\_id = 'SA\_REP'

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

rollback ;

--12. Schimbaţi salariul şi comisionul celui mai prost plătit salariat din firmă, astfel încât să fie egale cu salariul si comisionul şefului său.

update emp\_bai

set (salary, commission\_pct) = ( select salary, commission\_pct

from emp\_bai

where manager\_id is not null and last\_name = 'King')

where salary = (select min(salary) from emp\_bai )

--18. Să se modifice adresa de e-mail pentru angajaţii care câştigă cel mai mult în departamentul în care lucrează astfel încât acesta să devină iniţiala numelui concatenată cu prenumele. Dacă nu are prenume atunci în loc de acesta apare caracterul ‘.’. Anulaţi modificările.

update emp\_bai e

set email = substr (last\_name, 1, 1) || substr(first\_name) || '@labbd'

where salary = (select max(salary)

from emp\_bai

where department\_id = e.department\_id);

SAU

update emp\_bai e

set email = substr (last\_name, 1, 1) || substr(first\_name) || '@labbd'

where (department\_id, salary) in (select department\_id, max(salary)

from emp\_bai

group by department\_id);

rollback;

--19. Pentru fiecare departament să se mărească salariul celor care au fost angajaţi primii astfel încât să devină media salariilor din companie. Ţineţi cont de liniile introduse anterior.

update emp\_bai

set salary = (select avg(salary) from emp\_bai), commission\_pct = 0.5

where hire\_date = (select min(hire\_date) from emp\_bai)

--22. Ştergeţi toate înregistrările din tabelul DEPT\_PNU. Ce înregistrări se pot şterge?

-- constrangerile de FK nu ne lasa sa stergem campurile care contribuie la alte tabele

delete dept\_bai

--24. Suprimaţi departamentele care un au nici un angajat. Anulaţi modificările.

!!! daca folosim NOT IN -> nu trb sa avem campuri nule, pt ca asa ar fi eronat rezultatul

---->>>> TRB SA FOL SI is not nul pt a rezolva eroarea

delete dept\_bai

where department\_id NOT IN (select department\_id

from emp\_bai

where department\_id is not null)

--pus intr un script separat pt rulare.... (cam optional nu ti bate capul cu accept )

ACCEPT p\_cod PROMPT "cod= ";

SELECT employee\_id, last\_name, salary, department\_id

FROM employees WHERE employee\_id = &p\_cod;

-------------

--27 Să se marcheze un punct intermediar in procesarea tranzacţiei.

rollback;

delete dept\_bai2

where department\_id

not in (select department\_id

from emp\_bai

where department\_id is not null)

savepoint a;

delete dept\_bai2;

rollback to a; --- ma intorc pana la a