

Baze de date-Anul 1

Solutii – Lab4

-- LABORATOR 4 (continuare)

-- SUBCERERE

```
SELECT -- cerere parinte
FROM
WHERE col = (SELECT col
              FROM
              WHERE ... ); -- cerere copil sau cerere imbricata
```

12. Măriți salariul TUTUROR angajaților din tabelul EMP_PNU cu 5%.
Vizualizați, iar apoi anulați modificările.

```
SELECT * FROM EMP;
```

```
UPDATE emp
SET salary = salary * 1.05;
```

```
SELECT * FROM emp;
```

```
ROLLBACK;
```

13. Schimbați jobul tuturor salariaților din departamentul 80 care au comision, în 'SA_REP'.
Anulați modificările.

```
SELECT *
FROM EMP
WHERE DEPARTMENT_ID = 80 and commission_pct IS NOT NULL;
```

```
UPDATE emp
SET job_id = 'SA_REP'
WHERE department_id = 80 and commission_pct IS NOT NULL;
```

```
SELECT * FROM emp;
```

```
ROLLBACK;
```

14. Să se promoveze Douglas Grant la manager în departamentul 20, având o creștere de salariu cu 1000\$.

```
SELECT * FROM dept;
```

-- se foloseste doar pentru verificare

```
SELECT *
FROM EMP
WHERE employee_id = 201;
```

```
UPDATE dept
SET manager_id = (SELECT employee_id
                  FROM emp
                  WHERE lower(last_name || first_name) = 'grantdouglas');
```

```
ROLLBACK;
```

```

UPDATE dept
SET manager_id = (SELECT employee_id
                  FROM emp
                  WHERE lower(last_name || first_name) = 'grantdouglas')
WHERE department_id = 20;

-- creștere de salariu cu 1000$
SELECT * FROM dept;

-- se foloseste doar pentru verificare
select * from emp
where employee_id = 199;

-- rezolvarea cerintei
UPDATE emp
SET salary = salary + 1000
WHERE lower(last_name || first_name) = 'grantdouglas';

--modificam departamentul lui grant
UPDATE emp
SET department_id = (SELECT department_id
                    FROM dept
                    WHERE manager_id = (SELECT employee_id
                                        FROM EMP
                                        WHERE lower(first_name || last_name) = 'douglasgrant'
                                        )
                    )
WHERE lower(first_name || last_name) = 'douglasgrant';

```

-- Ce se intampla daca executam comanda ROLLBACK?

ROLLBACK; -- anuleaza modificarile

15. Ștergeți toate înregistrările din tabelul DEPT_PNU.

Ce înregistrări se pot șterge?

Anulați modificările.

DELETE FROM dept; -- cheia primara din tabelul dept este cheie externa in tabelul emp

-- ceea ce inseamna ca nu putem sterge id-ul departamentului din dept

-- daca acest id se afla pe post de cheie externa in emp

-- => daca vrem sa stergem din dept, trebuie sa stergem prima data din tabelul de legatura
- din emp

SELECT * FROM DEPT;

SELECT * FROM EMP;

-- se pot sterge doar departamentele care nu au angajati

DELETE FROM dept

WHERE department_id NOT IN (SELECT department_id

FROM emp

WHERE department_id is not null);

-- codul de departament este selectat din emp

-- => obtinem departamentele care au angajati

-- afisam departamentele care nu au angajati

SELECT department_id

FROM dept

WHERE department_id NOT IN (SELECT department_id

```
FROM emp
WHERE department_id is not null);
```

--SAU:

```
SELECT department_id
FROM dept
WHERE department_id NOT IN (SELECT NVL(department_id, -1)
                             FROM emp
                             );
```

```
SELECT * FROM DEPT;
```

```
ROLLBACK;
```

-- stergerea departamentelor care nu au angajati -> FOLOSIND IN

```
DELETE FROM dept
WHERE department_id IN (SELECT d.department_id
                        FROM emp e RIGHT JOIN dept d ON (e.department_id = d.department_id)
                        WHERE e.employee_id IS NULL);
```

-- subcererea trebuie sa returneze departamentele care nu au angajati

```
ROLLBACK;
```

-- se poate utiliza in subcerere si MINUS - TODO

17. Să se mai introducă o linie in tabelul DEPT_PNU.

```
desc dept;
```

```
INSERT INTO dept
```

```
VALUES(320, 'dept_nou', NULL, NULL);
```

```
SELECT * FROM dept;
```

18. Să se marcheze un punct intermediar in procesarea tranzacției (SAVEPOINT p).

```
SAVEPOINT p;
```

19. Să se șteargă din tabelul DEPT_PNU departamentele care au codul de departament cuprins intre 160 si 200 .

Listați conținutul tabelului.

```
DELETE FROM dept
```

```
WHERE department_id BETWEEN 160 AND 200;
```

20. Să se renunțe la cea mai recentă operație de ștergere, fără a renunța la operația precedentă de introducere.

(ROLLBACK TO p). Determinați ca modificările să devină permanente;

```
SELECT * FROM DEPT;
```

```
ROLLBACK TO p;
```

```
COMMIT;
```