

## Recapitulare SQL

---

1. Indicați valoarea de adevăr a următoarelor afirmații:
  - 1) Comenzile *SQL\*Plus* accesează baza de date. F
  - 2) Funcțiile grup se aplică asupra unei mulțimi de înregistrări și întorc un singur rezultat. A
  - 3) Funcțiile grup includ în calcule valorile *null*. F
2. Alegeți afirmația adevărată:
  - a. Cheia primară a unei tabele nu poate fi dezactivată ulterior adăugării ei, ci doar eliminată.
  - b. O tabelă poate avea declarată o singură constrângere de cheie primară.
  - c. Cheia primară a unei tabele nu poate fi compusă din mai multe coloane ale acestuia.
  - d. Pentru a putea adăuga o constrângere de cheie primară pe o coloană a unei tabele, coloana respectivă trebuie să conțină valori fără duplicate sau valori *null*.

B

3. Alegeți afirmația incorectă:
  - a. Constrângerea de cheie externă implementează o relație de tip one-to-many între două tabele.
  - b. Constrângerea de cheie externă se adaugă tablei “copil” și trebuie să refere o cheie unică sau primară din tabela “părinte”.
  - c. Ștergerea unei linii din tabela “părinte” implică întotdeauna ștergerea liniilor corespunzătoare acesteia din tabela “copil”, dacă relația dintre cele două tabele este implementată cu ajutorul unei constrângeri de cheie externă.
  - d. Coloana din tabela “copil” pe care este declarată o constrângere de cheie externă poate conține valori *null* sau valori menținute în coloana referită din tabela “părinte”.

C

4. O constrângere de validare
  - a. poate fi declarată doar la crearea tablei.
  - b. poate fi declarată doar ulterior creării tablei.
  - c. definește explicit o condiție ce trebuie satisfăcută doar de anumite linii ale tablei.
  - d. definește explicit o condiție ce trebuie satisfăcută de fiecare linie a tablei.

D

5. O vizualizare simplă (extrage date dintr-o singură tabelă, nu conține funcții și grupări de date)
  - a. nu reflectă întotdeauna actualizările realizate asupra tablei de bază.
  - b. stochează datele obținute prin execuția cererii din definiția ei.

## Recapitulare SQL

---

- c. determină ștergerea unei linii din tabela de bază, atunci când linia respectivă este ștearsă din vizualizare.
- d. nu permite actualizarea tabelului de bază prin intermediul său.

C

6. O subcerere care întoarce cel puțin două linii nu poate fi utilizată într-o comandă SELECT în clauza
- a. SELECT
  - b. FROM
  - c. WHERE
  - d. HAVING

A

7. Execuția comenzii următoare

```
SELECT titlu
FROM carte
WHERE cod_autor NOT IN (SELECT id_autor
                        FROM autor
                        WHERE nationalitate = 'Romana');
```

determină execuția subcererii sale de un număr de ori egal cu

- a. 1
- b. 0
- c. numărul de autori de naționalitate Română din tabelul “autor”
- d. numărul de linii din tabelul “carte”

A

8. Dacă în tabela “angajat” sunt menținute informații despre angajați, respectiv despre departamentul și jobul pe care lucrează în prezent, iar în tabela “istoric\_angajat” informații despre departamentele și joburile pe care au lucrat aceștia în trecut, atunci comanda următoare

```
SELECT id_angajat, cod_departament, cod_job
FROM angajat
INTERSECT
SELECT cod_angajat, cod_departament, cod_job
FROM istoric_angajat;
```

## Recapitulare SQL

---

obține angajații care în prezent lucrează

- într-un departament în care au lucrat și în trecut.
- pe un job pe care au lucrat și în trecut.
- în același departament și pe același job pe care au lucrat și în trecut.
- într-un departament și pe un job pe care nu au mai lucrat în trecut.

C

9. Se dau următoarele trei tabele:

```
FACTURA(id_factura#, data_facturare)
CONTINE(cod_factura#, cod_produs#, cantitate)
PRODUS(id_produs#, denumire, pret_unitar)
```

Comanda următoare

```
SELECT cod_factura, SUM(cantitate*pret_unitar)
FROM contine a, produs b, factura c
WHERE a.cod_produs = b.id_produs
AND a.cod_factura = c.id_factura
AND TO_CHAR(data_facturare, 'yyyy') = TO_CHAR(sysdate,
'yyyy')
GROUP BY cod_factura;
```

obține

- valoarea totală a tuturor facturilor emise în anul curent.
- valoarea totală a fiecărei facturi emise la o dată egală cu data curentă.
- valoarea totală a fiecărei facturi emise în anul curent.
- valoarea totală a tuturor facturilor emise la o dată egală cu data sistemului.

C

10. Se dă următoarea tabelă:

```
STUDENT(id_student#, nume, prenume, an_nastere, oras,
cod_camin);
```

Comanda următoare

```
INSERT INTO student
VALUES (100, 'Popescu', 'Andrei', 1980, 'Bucuresti');
```

- determină adăugarea unei linii în tabela “student” cu informațiile date.
- determină adăugarea unei linii în tabela “student” cu informațiile date, iar pentru coloana “cod\_camin” utilizează valoarea *null*.

## Recapitulare SQL

---

- c. determină adăugarea unei linii în tabela “student” cu informațiile date, iar pentru coloana  
“cod\_camin” utilizează valoarea *null* doar dacă această coloană nu are definită o  
valoare implicită.
- d. nu are efect deoarece se termină cu o eroare.

D

11. Adăugați un comentariu tablei *emp\_\*\*\**.

```
COMMENT ON TABLE emp_*** IS 'Informații despre angajati';
```

12. Folosind vizualizarea *user\_tab\_comments* afișați comentariul adăugat tabelului *emp\_\*\*\**.

```
SELECT *  
FROM   user_tab_comments  
WHERE  table_name = upper('emp_***');
```

13. Modificați formatul datei calendaristice setat la nivel de sesiune astfel încât datele  
calendaristice să respecte următoarea formă 01.10.2011 16:10:05.

*Indicație:* Folosiți comanda

```
ALTER SESSION SET NLS_DATE_FORMAT = 'formatul dorit';
```

```
ALTER SESSION
```

```
SET NLS_DATE_FORMAT = 'DD.MM.YYYY HH24:MI:SS';
```

14. Rulați următoarea cerere SQL:

```
SELECT EXTRACT(YEAR FROM SYSDATE)  
FROM dual;
```

15. Modificați cererea anterioară astfel încât să obțineți ziua, respectiv luna datei curente.

16. Afișați numele tuturor tabelelor personale create (nume\_tabel\_\*\*\*).

*Indicație:* Folosiți vizualizarea *user\_tables*.

```
SELECT table_name  
FROM   user_tables  
WHERE  table_name LIKE UPPER('%_***');
```

17. Generați automat un script SQL care să conțină comenzi de ștergere a tuturor tabelelor  
personale create. *Indicație:* Folosiți comenzile SPOOL .../sterg\_tabele.sql și SPOOL OFF.  
spool c:/sterg\_tabele.sql

```
SELECT 'DROP TABLE ' || table_name || ';'   
FROM   user_tables  
WHERE  table_name LIKE upper('%_***');  
spool off
```

18. Verificați informațiile din fișierul generat.

19. Ce informații suplimentare sunt incluse în acest fișier dacă folosim SQL\*Plus?

## Recapitulare SQL

---

```
SQL> SELECT 'DROP TABLE ' || table_name || ';'
2   FROM      user_tables
3   WHERE      table_name LIKE upper('%_test');

'DROPTABLE' || TABLE_NAME || ';'
-----
DROP TABLE CARTE_TEST;
DROP TABLE CITITOR_TEST;
DROP TABLE DEPARTAMENT_TEST;
DROP TABLE DEPARTMENT_TEST;
DROP TABLE DEPT_TEST;
DROP TABLE DOMENIU_TEST;
DROP TABLE ECO_TEST;
DROP TABLE EMP0_TEST;
DROP TABLE EMP1_TEST;
DROP TABLE EMP2_TEST;
DROP TABLE EMP3_TEST;

'DROPTABLE' || TABLE_NAME || ';'
-----
DROP TABLE EMP_TEST;
DROP TABLE IMPRUMUTA_TEST;
DROP TABLE PROJECTS_TEST;
DROP TABLE SALARIAT_TEST;
DROP TABLE WORK_TEST;

16 rows selected.

SQL> spool off
```

20. Verificați ce efect are utilizarea comenzii SET FEEDBACK OFF. Set feedback off

```
spool c:/sterg_tabele.sql

SELECT 'DROP TABLE ' || table_name || ';'
FROM      user_tables
WHERE      table_name LIKE upper('%_***');

spool off
set feedback on
```

21. Asigurați-vă că antetul tabelului rezultat nu se multiplică.

*Indicație:* Utilizați comanda SET PAGESIZE 0

## Recapitulare SQL

---

```
set pagesize 0
set feedback off
spool c:/sterg_tabele.sql

SELECT 'DROP TABLE ' || table_name || ';'
FROM   user_tables
WHERE  table_name LIKE upper('%_***');

spool off
set feedback on
set pagesize 10
```

22. Fără să rulați scriptul creat dați exemplu de un caz în care execuția acestui script va determina erori. Indicați o metodă de rezolvare a acestui caz.
23. Folosind tabelul *departments* generați automat script-ul SQL de inserare a înregistrărilor în acest tabel.

```
SELECT
    'INSERT INTO departments VALUES
    (' || department_id || ', ''' || department_name ||
    ''', ' || location_id || ');'
AS "Inserare date"
FROM   departments;
```

## TEMĂ

1. Identificați diagrama conceptuală definită în proiectul prezentat la materia Baze de Date din anul I. Dacă nu aveți o astfel de diagramă, atunci definiți un model simplificat pentru acest exercițiu.
  - a. adăugați ca poză diagrama conceptuală identificată;
  - b. adaptați cerințele exercițiilor 17 și 23 pentru diagrama conceptuală utilizată la punctul a (formulați cerința în limbaj natural, apoi rezolvați cererea propusă în SQL).