

**LABORATOR SQL RECAPITULARE – 1**

1. Indicați valoarea de adevăr a următoarelor afirmații:
  - a. Comenzile SQL\*Plus accesează baza de date. F
  - b. Funcțiile grup se aplică asupra unei mulțimi de înregistrări și întorc un singur rezultat. A
  - c. Funcțiile grup includ în calcule valorile null. F
2. Indicați valoarea de adevăr a următoarelor afirmații:
  - a. Cheia primară a unei tabele nu poate fi dezactivată ulterior adăugării ei, ci doar eliminată. F
  - b. O tabelă poate avea declarată o singură constrângere de cheie primară. A
  - c. Cheia primară a unei tabele nu poate fi compusă din mai multe coloane ale acestuia. F
  - 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*. F
3. Indicați valoarea de adevăr a următoarelor afirmații:
  - a. Constrângerea de cheie externă implementează o relație de tip one-to-many între două tabele. F
  - b. Constrângerea de cheie externă se adaugă tablei “copil” și trebuie să refere o cheie unică sau primară din tabela “părinte”. F
  - c. Ștergerea unei linii din tabela “părinte” implică întotdeauna ștergerea liniilor corespunzătoare acestuia din tabela “copil”, dacă relația dintre cele două tabele este implementată cu ajutorul unei constrângeri de cheie externă. A
  - 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”. F
4. O constrângere de validare
  - a. poate fi declarată doar la crearea tablei. F
  - b. poate fi declarată doar ulterior creării tablei. F
  - c. definește explicit o condiție ce trebuie satisfăcută doar de anumite linii ale tablei. F
  - d. definește explicit o condiție ce trebuie satisfăcută de fiecare linie a tablei. A
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. F
  - b. stochează datele obținute prin execuția cererii din definiția ei. F
  - c. determină ștergerea unei linii din tabela de bază, atunci când linia respectivă este ștearsă din vizualizare. A
  - d. nu permite actualizarea tablei de bază prin intermediul său. F
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
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”
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;
```

obține angajații care în prezent lucrează

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

9. Se dau următoarele trei tabele:

```
FACTURA(id_factura#, data_facturare)
CONTINE(cod_factura#, cod_produș#, cantitate)
PRODUS(id_produș#, denumire, pret_unitar)
```

Comanda următoare

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

obține

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

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 ');
```

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

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\_\*\*\**.

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';
```

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*.

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*.

18. Verificați informațiile din fișierul generat.

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

20. Verificați ce efect are utilizarea comenzii *SET FEEDBACK OFF*.

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

*Indicație:* Utilizați comanda *SET PAGESIZE 0*

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.