

LABORATOR 1 SQL

CERERI MONOTABEL

1. Ce este SQL?

- SQL (Structured Query Language) este un limbaj neprocedural pentru interogarea și prelucrarea informațiilor din baza de date.
- Limbajul SQL permite atât definirea, prelucrarea și interogarea datelor, cât și controlul accesului la acestea.
- Comenzile SQL pot fi integrate în programe scrise în alte limbaje, de exemplu C/C++, Java etc.

2. Care sunt limbajele SQL?

În funcție de tipul acțiunii pe care o realizează, instrucțiunile SQL se împart în mai multe categorii:

- limbajul de definire a datelor (LDD) – comenzile CREATE, ALTER, DROP;
- limbajul de prelucrare a datelor (LMD) – comenzile INSERT, UPDATE, DELETE, SELECT;
- limbajul de control al datelor (LCD) – comenzile COMMIT, ROLLBACK.

3. Comenzile SQL acceptă abrevieri? Necesită un caracter de terminare a comenzii?

- Comenzile SQL nu acceptă abrevieri.
- Caracterul de terminare al unei comenzi SQL este “;”.

4. Ce este SQL*Plus? Comenzile SQL*Plus accesează baza de date?

- SQL*Plus este un utilitar Oracle, având comenzi proprii.
- Este o interfață specifică sistemului Oracle pentru execuția instrucțiunilor SQL.
- Nu permite prelucrarea informațiilor din baza de date.

5. Comenzile SQL*Plus acceptă abrevieri? Necesită un caracter de terminare a comenzii?

- Comenzile pot fi abreviate.
- Nu sunt necesare caractere pentru a specifica încheierea comenzii.
- Acceptă „–” drept caracter de continuare pentru comenzile scrise pe mai multe linii.

6. Analizați sintaxa simplificată a comenzii SELECT. Care dintre clauze sunt obligatorii?

```
SELECT { [ {DISTINCT | UNIQUE} | ALL] lista_campuri | *}
FROM [nume_schemă.]nume_obiect  ]
      [, [nume_schemă.]nume_obiect  ...]
[WHERE condiție_clauza_where]
[[START WITH condiție_clauza_start_with]
  CONNECT BY condiție_clauza_connect_by]
[GROUP BY expresie [, expresie ...]
  [HAVING condiție_clauza_having] ]
[ORDER BY {expresie | poziție} [, {expresie | poziție} ...] ]
```

[**FOR UPDATE**

```
[OF [ [nume_schemă.]nume_obiect.]nume_coloană  
[, [ [nume_schemă.]nume_obiect.]nume_coloană] ...]  
[NOWAIT | WAIT număr_întreg] ];
```

7. Inițiați o sesiune SQL*Plus folosind informațiile de conectare indicate.

8. Găsiți erorile din instrucțiunea următoare.

```
SELECT employee_id, last_name  
       salary * 12 salariu anual  
FROM employees;
```

Observație:

- SALARIU ANUAL este un alias pentru câmpul care reprezintă salariul anual.
- Dacă un alias conține *blank*-uri, acesta va fi scris obligatoriu între ghilimele. Altfel, ghilimelele pot fi omise.
- *Alias*-ul apare în rezultat, redenumind coloana corespunzătoare expresiei respective.
- *Alias*-urile specificate între ghilimele sunt *case-sensitive*, celelalte fiind scrise implicit cu majuscule.

Varianta 1:

```
SELECT employee_id, last_name, salary * 12 salariu_anual  
FROM employees;
```

Varianta 2:

```
SELECT employee_id, last_name, salary * 12 "Salariu Anual"  
FROM employees;
```

9. a) Consultați diagrama exemplu HR (Human Resources) pentru lucrul în cadrul laboratoarelor SQL.
b) Identificați cheile primare și cele externe ale tabelelor existente în schemă, precum și tipul relațiilor dintre aceste tabele.

10. Listați structura tabelelor din schema HR, observând tipurile de date ale coloanelor.

Observație: Se va utiliza comanda SQL*Plus

```
DESCRIBE nume_tabel
```

11. Listați conținutul tabelelor din schema considerată, afișând valorile tuturor câmpurilor.

Observație: Se va utiliza comanda SQL

```
SELECT * FROM nume_tabel;
```

12. Obțineți încă o dată rezultatul cererii precedente, fără a rescrie cererea.

Observație: Ultima comandă SQL lansată de către client este păstrată în buffer-ul SQL.

Pentru rularea acesteia se utilizează “/” sau RUN.

13. Listați structura tabelului EMPLOYEES și apoi utilizați comanda RUN (sau “/”). Ce observați? Comenzile SQL*Plus sunt păstrate în buffer?

14. Afișați codul și numele angajatului, respectiv codul job-ului și data angajării. Salvați instrucțiunea SQL într-un fișier numit p1_14.sql.

Observație: Pentru salvarea ultimei comenzi SQL se utilizează comanda SAVE.

Precizarea extensiei „.sql” a fișierului nu este obligatorie.

```
SAVE z:\...\p1_14.sql
```

15. Reexecutați cererea folosind fișierul p1_14.sql.

```
START z:\...\ p1_14.sql
```

sau

```
@ z:\...\ p1_14.sql
```

16. Editați fișierul p1_14.sql, adăugând coloanelor câte un alias (cod, nume, cod job, data angajării).

```
EDIT z:\...\ p1_14.sql
```

17. Listați, cu și fără duplicate, codurile job-urilor din tabelul EMPLOYEES.

Observație: DISTINCT = UNIQUE

18. Pentru fiecare angajat afișați numele acestuia concatenat cu prenumele său. Redenumiți coloana “Nume si prenume”.

Observație: Operatorul de concatenare este “||”. Șirurile de caractere se specifică între apostrofuri (**NU** ghilimele, caz în care ar fi interpretate ca alias-uri).

19. Afișați numele și salariul angajaților care câștigă mai mult de 10000 \$.

20. Modificați cererea anterioară astfel încât să afișați numele și salariul pentru toți angajații al căror salariu este cuprins între 5000\$ și 10000\$.

Observație: Pentru testarea apartenenței la un domeniu de valori se poate utiliza operatorul

```
[NOT] BETWEEN valoare1 AND valoare2
```

21. Afișați numele și salariul angajaților din departamentele 10 și 30, în ordine alfabetică a numelor.

Observație: Apartenența la o mulțime finită de valori se poate testa prin intermediul operatorului IN, urmat de lista valorilor între paranteze și separate prin virgule:

```
expresie IN (valoare_1, valoare_2, ..., valoare_n)
```

22. Listați numele și salariile angajaților care câștigă mai mult de 10000 \$ și lucrează în departamentul 10 sau 30. Etichetați coloanele cu „Angajat” și „Salariu lunar”.

23. Care este data curentă?

Observație:

- Funcția SYSDATE întoarce data curentă și timpul (ore, minute, secunde) a sistemului de operare pe care se află server-ul de baze de date.
- Funcția CURRENT_DATE este echivalentă funcției SYSDATE, numai că întoarce data setată la nivel de sesiune.
- Funcția SYSTIMESTAMP întoarce data curentă, timpul (ore, minute, secunde și fracțiuni de secunde) și diferența de fus orar față de ora universală a sistemului de operare pe care se află baza de date.
- Funcția CURRENT_TIMESTAMP este echivalentă funcției SYSTIMESTAMP, numai că întoarce data setată la nivel de sesiune.
- Funcția LOCALTIMESTAMP este echivalentă funcției CURRENT_TIMESTAMP, numai că nu conține și diferența de fus orar față de ora universală.
- Pentru completarea sintaxei obligatorii a comenzii SELECT, se utilizează tabelul DUAL:

```
SELECT SYSDATE, CURRENT_DATE
FROM dual;
```

```
SELECT SYSTIMESTAMP, CURRENT_TIMESTAMP, LOCALTIMESTAMP
FROM dual;
```

24. Afișați data sistemului într-un format diferit de cel implicit.

Observație: Datele calendaristice pot fi formatate cu ajutorul funcției TO_CHAR(data, 'format'), unde formatul poate fi alcătuit dintr-o combinație a următoarelor elemente:

Element	Semnificație
D	Numărul zilei din săptămână (duminică=1; luni=2; ...sâmbătă=7).
DD	Numărul zilei din lună.
DDD	Numărul zilei din an.
DY	Numele zilei din săptămână, printr-o abreviere de 3 litere (MON, THU etc.)
DAY	Numele complet al zilei din săptămână.
MM	Numărul lunii din an.
MON	Numele lunii din an, printr-o abreviere de 3 litere (JAN, FEB etc.).
MONTH	Numele complet al lunii din an.
Y	Ultima cifră din an.
YY, YYYY	Ultimele 2, respectiv 3 cifre din an.
YYYY	Anul în format numeric (4 cifre).
YEAR	Anul, scris în cuvinte (ex: two thousand four).
W	Numărul săptămânii din lună
WW	Numărul săptămânii din an

Q	Numărul trimestrului din an
HH12, HH24	Orele din zi, între 0-12, respectiv 0-24.
AM sau PM	Indicator meridian: AM sau PM
MI	Minutele din oră.
SS	Secunde din minut.
SSSSS	Secunde trecute de la miezul nopții.
CC	Secolul

25. Afișați numele și data angajării pentru fiecare salariat care a fost angajat în 1987. Se cer 2 soluții: una în care se lucrează cu formatul implicit al datei și alta prin care se formează data.

Varianta 1:

```
SELECT first_name, last_name, hire_date
FROM employees
WHERE hire_date LIKE ('%87');
```

Varianta 2:

```
SELECT first_name, last_name, hire_date
FROM employees
WHERE TO_CHAR(hire_date, 'YYYY') = '1987';
```

Sunt obligatorii apostrofurile utilizate în partea dreaptă a condiției ('1987')? Ce observați?

26. Afișați numele, job-ul și data la care au început lucrul salariații angajați în perioada 10 Februarie 1987 - 10 Februarie 1990. Rezultatul va fi ordonat crescător după data angajării.

```
SELECT last_name, job_id, hire_date
FROM employees
WHERE hire_date BETWEEN '10-FEB-1987' AND '10-FEB-1990'
ORDER BY hire_date;
```

27. Afișați numele și job-ul pentru toți angajații care nu au manager.

```
SELECT last_name, job_id
FROM employees
WHERE manager_id IS NULL;
```

28. Afișați numele, salariul și comisionul pentru toți salariații care câștigă comision. Să se sorteze datele în ordine descrescătoare a salariilor, iar pentru cei care au același salariu în ordine crescătoare a comisioanelor.

```
SELECT last_name, salary, commission_pct
FROM employees
WHERE commission_pct IS NOT NULL
ORDER BY salary DESC, commission_pct ASC;
```

29. Eliminați clauza WHERE din cererea anterioară. Unde sunt plasate valorile NULL în ordinea crescătoare?

30. Listați numele tuturor angajaților care au a treia litera din nume 'a'.

Observație: Pentru a forma măștile de caractere utilizate împreună cu operatorul LIKE cu scopul de a compara șirurile de caractere, se utilizează:

- caracterul „%” care reprezintă orice șir de caractere, inclusiv șirul vid;
- caracterul „_” care reprezintă un singur caracter.

```
SELECT DISTINCT last_name
FROM employees
WHERE last_name LIKE '__a%';
```

31. Folosind data curentă afișați următorul mesaj: „Suntem în Anul Luna ... Ziua ...”.

```
SELECT 'Suntem în' ||
       TO_CHAR(sysdate, ' "Anul" YYYY "Luna" fmMONTH "Ziua" DD') data
FROM DUAL;
```

32. Folosind data curentă afișați următoarele informații:

- numele zilei, numărul zilei din săptămână, numărul zilei din luna, respectiv numărul zilei din an;
- numărul lunii din an, numele lunii cu abreviere la 3 caractere, respectiv numele complet al lunii;
- ora curentă (ora, minute, secunde).

33. Listați numele departamentelor care funcționează în locația având codul 1700 și al căror manager este cunoscut.

34. Afișați codurile departamentelor în care lucrează salariați.

35. Afișați numele și prenumele salariaților angajați în luna iunie 1987.

36. Listați codurile angajaților care au avut și alte joburi față de cel prezent. Ordonați rezultatul descrescător după codul angajatului.

37. Afișați numele și data angajării pentru cei care lucrează în departamentul 80 și au fost angajați în luna martie a anului 1997.

38. Afișați numele joburilor care permit un salariu mai mare de 8000\$.

39. Care este grila de salarizare a unui salariu de 10000\$?

40. Listați numele tuturor angajaților al căror nume conține 2 litere 'L'.

41. Afișați informații complete despre subordonații direcți ai angajatului având codul 123.

42. Afișați numele, salariul, comisionul și venitul lunar total pentru toți angajații care câștigă comision, dar un comision care nu depășește 25% din salariu.