

**01. Explicați diferențele între:**

```
SELECT b
FROM R
WHERE a<10 or a>=10;
```

Si

```
SELECT b
FROM R;
```

? Daca a are valoarea NULL atunci prima varianta nu va afisa acea tupla, iar a doua o va afisa.

**02. Explicați diferențele între:**

```
SELECT a
FROM R, S
WHERE R.b = S.b;
```

si

```
SELECT a
FROM R
WHERE b IN (SELECT b FROM S);
```

A doua interogare returneaza doar o singura tupla din S in cazul in care sunt mai multe valori egale ale lui **b**, adica o valoare a lui **a** va fi afisat o singura data. In prima interogare, valoarea lui **a** va fi afisata de fiecare data cand se gaseste **R.b = S.b**, adica de mai multe ori. (Curs "02 SQL partea 2", slide 44)

**03. Ce domenii de aplicativitate acoperă bazele de date?**

În zilele noastre, domeniul BD acoperă largi surse de date, cu multe concepte noi.

Căutare Web.

Data mining.

Baze de Date Științifice și medicale.

Integrarea informației.

**04) ce este o baza de date?**

O baza de date este o colectie in general mare de date inrudite stocate intr-un sistem de calcul astfel incat un program-calculator sau o persoana ce foloseste un limbaj de interogare sa o poata consulta pentru a raspunde la intrebari.

**05. Limbajele de programare pentru baze de date sunt complete (în sensul masinii Turing)? Explicați.**

Programare în context BD

Isi gasesc sens doar in aria unde se pot utiliza limbaje "non -Turing-complete"

**06. Ce probleme apar în contextul bazelor de date?**

Bazele de Date generează probleme unice ce țin de controlul concurenței.

Multe activități (tranzacții) cu baza de date se desfășoară la același moment de timp.

Nu trebuie încurcate acțiunile, de exemplu două extrageri de bani din același cont bancar trebuie fiecare să debiteze contul.

**07. Definiți ce este un model de date.**

- .O reprezentare matematică a datelor.

Exemple:

- modelul relațional = tabele;

- modelul semistructurat = arbori/grafuri.

- .Operații cu datele.

- .Constrângeri.

**08. Enunțați funcțiile unui Sistem de Gestiune de Baze de Date.**

- gestiunea dicționarului de date
- gestiunea fișierelor de date
- asigurarea integrității datelor
- transformarea datelor
- gestiunea aplicațiilor
- importul și exportul datelor
- controlul securității datelor
- controlul accesului concurrent la date
- gestiunea copiilor de siguranță și a recuperării datelor în caz de dezastre

**09. Câte nivele de abstractizare există la un Sistem de Gestiune de Baze de Date?**

Nivel extern, Nivel conceptual și Nivel intern

**10. Unde se folosește (în ce context, nivel de abstractizare) limbajul de descriere a datelor?**

Schema conceptuală sau schemă logică, descrie datele stocate în BD în termeni ai modelului de date al SGBD  
Pentru un SGBD relațional, schema conceptuală descrie toate tabelele (relațiile) stocate în BD  
Limbajul folosit se numește Limbaj de Descriere a Datelor (LDD)

**11. Unde se folosește (în ce context, nivel de abstractizare) limbajul de manipulare a datelor?**

Gestiunea aplicațiilor.

**12. Câte nivele de independență a datelor cunoașteți pentru un Sistem de Gestiune de Baze de Date?**

Aplicațiile sunt izolate față de modificările la nivel conceptual sau la nivel fizic prin cele trei nivele de abstractizare

Independența logică a datelor

Vederile (view în modelul relațional, tabelă virtuală, schema externă) asigură posibilitatea modificării structurii datelor (schema conceptuală), acest lucru fiind ascuns aplicațiilor

Independența fizică a datelor

Schema conceptuală asigură posibilitatea modificării aranjării datelor pe suport secundar sau a indecșilor, acest lucru fiind de asemenea ascuns aplicațiilor

**13. Descrieți pe scurt care este rolul optimizatorului în arhitectura unui Sistem de Gestiune de Baze de Date.**

Un utilizator emite o interogare, aceasta este analizată și este prezentată unui optimizator, care folosește informația despre felul în care sunt stocate datele pentru a produce un plan de execuție eficient

**14. Descrieți pe scurt categoriile de utilizatori ale unui Sistem de Gestiune de Baze de Date.**

Implementatori BD-Scriu software SGBD (soft de bază)

Programatori de Aplicații Bd-Dezvoltă pachete de programe ce facilitează accesul la date al utilizatorilor finali  
Folosesc "host languages" sau "data languages" și unele software

La modul ideal aplicațiile BD lucrează prin schema externă, dar este posibil să acceseze datele și la nivelele de mai jos când este posibil să fie compromisă independența datelor

Utilizatori finali-Înregistrează și interoghează BD, de obicei prin intermediul aplicațiilor

În marea majoritate a cazurilor nu sunt specialiști în domeniul calculatoarelor (de exemplu un agent comercial)

**15. Descrieți pe scurt arhitecturile pentru aplicațiile cu baze de date.**

- mainframe
- client – server
- N-Tier
- Web application

**16. Care este diferența între "thin clients" și "thick clients"?**

„thin clients”-La client există GUI

-Serverul se preocupă de logica aplicației și de accesul la BD

“thick clients”-Client mai puternic cu o parte din logica de aplicație și GUI

-Serverul are o parte din logica aplicației și accesul la BD

### 17. Din ce clauze este constituit blocul de bază în limbajul SQL (interogare)?

SELECT

FROM

WHERE (optional)

### 18. Enumerați operațiile de actualizare date în SQL

Adaugare element consola, modificare element din structura, si stergere element din structura

### 19. Explicați cum funcționează o interogare SQL pe o singură relație.

Forma cea mai simplă:

```
CREATE TABLE <nume> (  
    <listă de elemente>  
);
```

Pentru a elimina o relație:

```
DROP TABLE <nume>;
```

- relatia este specificata in clauza from

- se aplica selectia precizata in clauza where

- se aplica proiectia extinsa de lista de attribute si expresii din clauza select

### 20. Explicați cum funcționează o interogare SQL pe două relații.

Aprox. la fel ca și în cele pe o singură relație

1. se începe cu produsul relațiilor din clauza from

2. se aplica selectia impusa de conditia din clauza where

3 se aplica proiectia extinsa de lista de attribute si expresii din clauza select

### 21. Cum procedați pentru a redenumi attributele rezultat al unei interogări?

Dacă se dorește ca rezultatul să aibă nume diferite de attribute, se folosește “AS <nume nou>” pentru a redenumi un atribut.

Exemplu: Se folosește Beers(name, manf):

```
SELECT name AS beer, manf  
FROM Beers  
WHERE manf = 'Anheuser-Busch'
```

### 22. Explicați ce influență are existența valorilor NULL asupra unei interogări.

Dacă o valoare din expresia aritmetică este null, atunci întreaga expresie are valoarea null. În consecință tupla corespunzătoare nu va face parte din rezultat.

### 23. Dați un exemplu de OUTER-JOIN în SQL.

```
Select s.nr_matricol, s.num, s.prenume, n.nota
```

```
From student s outer join nota n
```

```
On s.nr_matr=n.nr_matr
```

### 24. Dați un exemplu de interogare imbricată.

Să se găsească berea preferată de cel puțin o persoană ce frecventează “Joe’s Bar”.

```
SELECT beer
```

```
FROM Likes, (SELECT drinker
```

```
FROM Frequent
```

```
WHERE bar = 'Joe's Bar')JD
```

WHERE Likes.drinker = JD.drinker;

**25. Poate fi utilizată o interogare imbricată ca si valoare?**

Da, in cazul in care aceasta returneaza o singura tupla.

**26. Ce restricție există pentru a utiliza operatorul SQL, UNION?**

Fiecare instructiune SELECT din cadrul operatorului UNION trebuie sa aiba acelasi nr de coloane. Coloanele trebuie sa aiba tipuri similare de date si sa fie în aceeași ordine.

**27.Cum pot fi eliminate tuplele duplicat din rezultatul unei interogări SQL?**

Prin folosirea operatorului DISTINCT.  
Sau Union, Intersect,Except

**28. Pot fi sortate tuplele rezultat ale unei interogări SQL?**

Da cu ajutorul lui ORDER BY nume [ASC/DESC]

**29.Ce restricții există pentru clauza SELECT (SQL) în cazul agregării?**

Daca se foloseste agregarea, atunci fiecare element din lista clauzei SELECT trebuie sa fie: o valoare Agregata sau un atribut al listei clauzei Group by.

**30.Explicați semanticile operației SQL DELETE.**

DELETE FROM <relatie>

WHERE <conditie>

Stergerea actioneaza in 2 etape:

1. semarcheaza toate tuplele pt care conditia where este satisfacuta
2. se eleimina tuplele marcate pt stergere

**31. Definiți ce este aceea constrângere în contextul bazelor de date.**

O constrangere este o relatie de legatura intre elemente de date pe care SGBD-ul este oblogat sa le forteze.

**32. Descrieți pe scurt tipurile de constrângeri în contextul bazelor de date.**

- chei
- cheie straina sau integritate-referentiala
- constrangeri la nivel de valoare - constrang valorile unui atribut particular
- constrangeri la nivel Tupla - relatie de lagatura intre componente

**33. Explicați care sunt violările posibile după definirea unei chei străine.**

Daca exista o constrangere cheie straina de la relatia r la relatia s, doua violari sunt posibile:

- la o adaugare (INSERT) sau modificare(UPDATE) a unei tuple in r se introduc valori ce nu se regasesc in s.
- dupa o stergere(DELETE) sau modificare(UPDATE) a unei tuple in s rezulta anumite tuple inconsecvente in r(nu au pereche in s)

**34. La o constrângere la nivel atribut condiția poate face referire la altă relație sau la alt atribut?**

Da conditia poate folosi numele atributului, dar orice alta relatie sau atribut ce apar trebuie folosite intro interogare imbricata.

**35. La ce operații sunt testate constrângerile la nivel atribut/tuplă?**

Cand se efectueaza o adaugare sau o modificare

**36. Explicați cum funcționează un trigger.**

Trigerele permit utilizatorului să decidă când să se verifice orice fel de condiție. Ele sunt compuse din 3 părți: eveniment, condiție și acțiune.

**37. Care este diferența între "row-level-triggers" și "statement-level-triggers"?**

Row level triggers se execută o dată pe fiecare tuplă modificată, iar statement-level triggers se execută o dată pe o instrucțiune SQL, indiferent de numărul tuplelor modificate.

**38. Definiți vederea SQL (view).**

O vedere este o relație definită în termeni de tabele stocate (numite tabele de bază) și alte vederi.

**39. Pot fi actualizate datele din baza de date prin intermediul unei vederi? Explicați răspunsul dat.**

Doar la folosirea unei vederi materializate.

Vederea restricționează accesul la date. Datele din tabelele de bază pot fi actualizate prin intermediul vederilor dacă se respectă urm. condiții:

- orice actualizare trebuie să facă referire la coloanele dintr-o singură tabelă de bază
- coloanele actualizate prin intermediul vederii trebuie să facă referire directă la coloanele din tabelă de bază. Coloanele nu pot fi coloane derivate (funcții de agregare, o expresie de calcul)
- nu este utilizată clauza `top n` nicăieri în instrucțiunea `select` a vederii combinată cu clauza `With Check OPTION`

**40. La ce este utilizat triggerul "INSTEAD OF"?**

Permite interpretarea actualizărilor produse vederii astfel încât actualizările să aibă sens

**41. Descrieți pe scurt relația existentă între PSM și SQL.**

- psm = un amalgam de instrucțiuni convenționale și sql
- oferă posibilități altfel inexistente în sql
- permite stocarea de proceduri ca elemente ale bazei de date.

**42. Ce este CLI?**

CLI (X/Open Call Level Interface) este un standard pe care JDBC se bazează și este implementat în interfata Microsoft ODBC

- este o unealtă de corectare care permite unui limbaj convențional să acceseze o bază de date

**43. Dați un exemplu de procedură PSM.**

```
create procedure Joemenu(  
in b char(20),  
in p real  
)  
insert into sells  
values('Joe's Bar',b,p);
```

**44. Dați un exemplu de funcție PSM.**

```
create function Suma(  
in b real,  
in p real,  
out s real  
)  
return a+b;
```

**45. Explicați diferența între RETURN în limbajul C și RETURN în PSM.**

Spre deosebire de C return în PSM nu termină execuția funcției

#### 46. Sub ce formă poate fi folosită o interogare SELECT-FROM-WHERE în PSM?

În general interogările select-from-where nu sunt permise în psm, însă există 3 moduri pentru a obține efectul unei interogări:

- interogările ce produc o singură valoare pot fi utilizate ca expresie într-o atribuire
- select....into având rezultat 1 tupla
- cursori

#### 47. Dați un exemplu de utilizare cursor în PSM.

```
create procedure joegourge()
declare berea char(20);
declare pretul real;
declare negasit condition for sqlstate '02000';
declare c cursor for
  (select beer, price from sells
   where bar = 'Joe's Bar');
begin
  open c;
  bucla_meniu: LOOP
    fetch c into berea,pretul;
    if negasit then leave bucla_meniu end if;
    if pretul<3.00 then
      update sells set price = pretul +1.00
      where bar='Joe's Bar' and beer = berea;
    end if;
  end loop;
  close c;
end;
```

#### 48. Dați un exemplu de precizare generică de tip în PL/SQL. La ce este utilă o astfel de precizare?

```
create or replace procedure JoeMenu(
  b in Sells.beer%type,
  p in Sells.beer%type
) as
begin
  insert into sells
  values('Joe's Bar',b,p);
end;
.
run
Sells.beer%type = precizare generica de tip...
```

#### 49. Care este mecanismul embedded SQL?

Un preprocesor translatează instrucțiunile SQL în apeluri de procedură ce se încadrează în codul limbajului gazdă.

#### 50. Explicați modul de utilizare SQL dinamic.

- pregătirea unei interogări: `exec sql prepare <nume-interogare>;`  
from <textul interogării>;
- execuția unei interogări: `Execc sql execute <nume-interogare>;`
- "prepare" = optimizare interogare
- se pregătește odată se execută de mai multe ori

**51. Explicați pe scurt utilizarea CLI din limbajul C.**

ajuta la corectarea bazelor de date a limbajelor conventionale prin intermediul bibliotecilor, biblioteca pt C este numita CALL-Level-Interface. In loc sa se foloseasca un preprocesor se poate folosi o biblioteca cu functii

**52. Explicați pe scurt utilizarea JDBC.**

- este o biblioteca similara cu sql/cli dar cu java ca limbaj gazda
- face distinctie intre interogari si actualizari

**53. Explicați diferența între ghilimele si apostrof în PHP.**

- ghilimelele semnifica "se inlocuieste"
- apostroful semnifica "nu se inlocuieste"

**54. Explicați cum se utilizează tablourile asociative în PHP.**

- elementele unui tablou asociativ \$a sunt perechi  $x \Rightarrow y$  unde x este un sir de caractere cheie si y este orice valoare
- daca  $x \Rightarrow y$  este un element al lui \$a, atunci  $a[x]$  este y

**55. Explicați pe scurt cum se accesează o bază de date din PHP (utilizând PEAR).**

exista biblioteca DB in pear si este inclusa cu include(DB.php.)