

## Subiect 4

[1] Enumerați funcțiile a unui SGBD.

- 1) Gestionarea dicționarului de date
- 2) Gestionarea fișierelor de date
- 3) Transformarea datelor
- 4) Gestionarea aplicațiilor
- 5) Importul și exportul datelor
- 6) Controlul securității datelor
- 7) Asigurarea integrității datelor
- 8) Controlul accesului concurent la date
- 9) Gestionarea copurilor de siguranță și a recuperării datelor în caz de dezastre

[2] Limbajul de descriere a datelor.

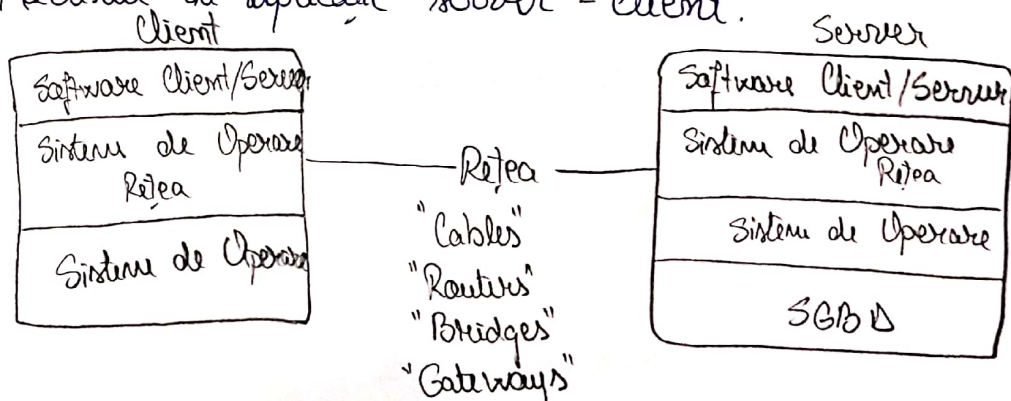
Este limbajul folosit pentru a descrie schema logică din nivelul conceptual al nivelului de abstractizare.

- DDL permite :
- crearea tabelelor
  - modificarea tabelelor ("alter")
  - ștergerea datelor.

3) Descrieți pe scurt categoriile de utilizatori pentru un SGBD.

- implementatori BD: scriu software SGBD
- 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 nivelurile de mai jos când este posibil să fie compromise independența datelor
- utilizatori finali:
  - înregistrează și interoghează BD, de obicei prin intermediul aplicațiilor
  - în marea majoritate a cazurilor nu sunt specialiști în domeniul calculatoarelor
- administratorul BD.

4) Modelul de aplicație server - client.



Funcții Client :

- Interfață grafică
- Procesare distribuită a aplicației
- Aplicație locală
- E-mail
- Emulare terminal

Funcții Server :

- Server : de fișiere, de tipărire la imprimantă  
și bază de date
- Procesare distribuită a aplicației
- E-mail
- Comunicații
- Administrare rețea
- Administrare resurse
- Administrare configurație

6) Dați exemple de utilizare cursor în PSM.

```
EXEC SQL BEGIN DECLARE SECTION;
```

```
    char Bona[21];
```

```
    float Pretul;
```

```
EXEC SQL END DECLARE SECTION;
```

```
EXEC SQL DECLARE c CURSOR FOR
```

```
    SELECT beer, price FROM Sells
```

```
    WHERE beer = 'Jai "s Bar";
```

```
EXEC SQL OPEN CURSOR c;
```

```
    while (1) {
```

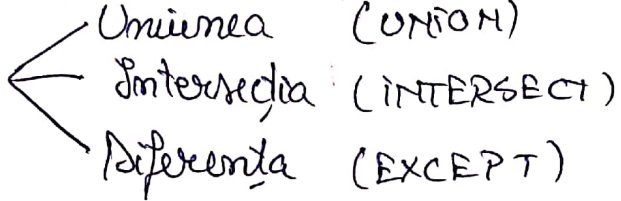
```
        EXEC SQL FETCH c
```

```
            INTO :Bona, :Pretul;
```

```
        if (NOT FOUND) break;
```

```
    } EXEC SQL CLOSE CURSOR c;
```

15 Dăți exemple de utilizare a operatorilor SQL „set” (pe multiple).

Operatori „set” 

- Unirea (UNION)
- Intersecția (INTERSECT)
- Diferența (EXCEPT)

ex: intersect: (SELECT \* FROM Likes)

INTERSECT

(SELECT drinker, beer  
FROM Sells, Frequent  
WHERE Frequent.beer = Sells.beer);

union: (SELECT \* FROM Likes)

UNION

(SELECT drinker, beer  
FROM Sells, Frequent  
WHERE Frequent.beer = Sells.beer);



II) Explicați pe scurt utilizarea JDBC.

JDBC ("Java Database Connectivity") este o bibliotecă similară cu SQL/CLT, dar cu Java ca limbaj gazdă.

JDBC oferă 2 clase: 1) Statement

2) PreparedStatement

Crearea instrucțiunilor:

Clasa "Connection" are metode pentru a crea "Statements" și "PreparedStatement".

Statement inst 1 = con - Mea. createStatement();

PreparedStatement inst 2 = con - Mea. createStatement()

"SELECT beer, price

FROM Sells" +

"WHERE bar = 'Joe's Bar'");

Execuția instrucțiunilor:

JDBC face distincție între interogări și actualizări, pe care le numește "updates".

"Statement" și "PreparedStatement" au fiecare metode:

- executeQuery
- executeUpdate

Pentru "Statement": există un argument - interogare sau actualizare ce trebuie executată.

Pentru "PreparedStatement": nici un argument.

care nu  
începe?

- 8) Ce proprietăți (restricții) trebuie să întrumească  
o formulă pt. a fi „sigură” în calculul relațional?

Proprietăți:

- Orice formulă sigură este formulă independentă de domeniu.
- Fînd dată o formulă  $F$ , se poate spune dacă este sau nu sigură.
- Orice interogare din algebra relațională poate fi exprimată cu ajutorul formulelor sigure.

- 9) Descrieți proprietățile unei descompunerii.

1) Coplarea fără pierdere de informații

- relațiile originale trebuie să poată fi obținute din schema descompusă, adică să se reconstituască originalul.

2) Conservarea DF-le:

- relațiile obținute prin descompunere trebuie să satisfacă toate DF-le inițiale.

Se poate obține (1) la o descompunere FNBC.

Se pot obține ambele (1) și (2) la o descompunere FNB.

Nu întotdeauna este posibil să se obțină ambele (1) și (2) la o descompunere FNBC.

Exemplul street-city-zip demonstrează acest fapt.

10) Care sunt caracteristicile pentru alegerea modelului NOSQL?

- 1) Analizarea datelor care vor fi stocate  
VOLUM MARE, VALOARE SCĂZUTĂ
- 2) Analizarea schemei aplicației  
DYNAMIC

## Subject 4

a) Sa se gaseasca nume, prenume, functie pt. profesori cu functie 'P' sau 'A'.

**SOL:** SELECT nume, prenume, functie  
FROM Profesor  
WHERE functie = 'L'

**APD:**  $\Pi_{\text{nume, prenume, functie}} \left( \sigma_{\text{functie} = 'L'} (\text{Profesor}) \right)$

**CPD:**  $\{ \langle B, C, D \rangle / \langle A, B, C, D, E \rangle \in \text{Profesor} \wedge D \neq 'L' \}$

**CRT:**  $\{ P.\text{nume}, P.\text{prenume}, P.\text{functie} / P \in \text{Profesor} \wedge P.\text{functie} \neq 'L' \}$

**Answer:**

$$\begin{array}{c} \Pi_{\text{nume, prenume, functie}} \\ \downarrow \\ \sigma_{\text{functie} \neq 'L'} \\ \downarrow \\ \text{Profesor} \end{array}$$



b) Sa se gasca nume profesor, prenume profesor, denumirea materiei, nota pt. examenul studentului cu nr-matn=14.

**SOL:**

SELECT p.nume, p.prenume, m.denumirea, e.nota  
FROM Profesor p, Examen e, Materie m  
WHERE e.nr-matn=14 and  
p.id-prof = e.id-prof and  
m.id-mat = e.id-mat;

**ARO:**

a := Profesor  $\bowtie$  Examen  $\bowtie$  Materie

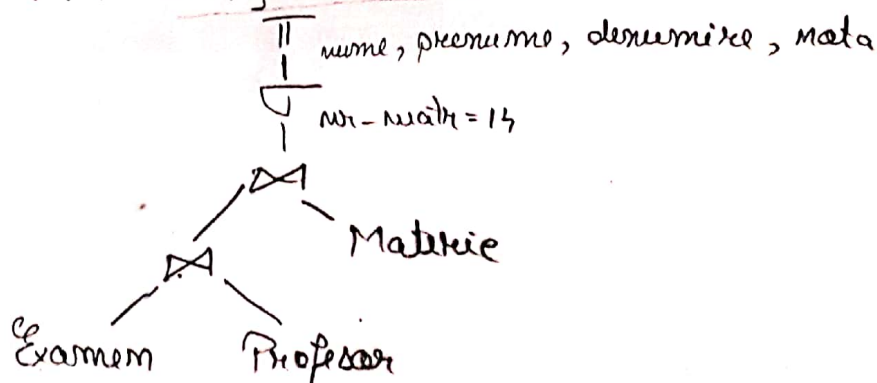
b :=  $\bigvee_{nr-matn=14} (a)$

rez :=  $\Pi_{nume, prenume, denumirea, nota}$  (b).

**CRT:**  $\{ \langle X.nume, X.prenume, X.denumirea, X.nota \rangle \mid \exists P \in Profesor \wedge \exists E \in Examen \wedge \exists M \in Materie (E.nr-matn=14 \wedge P.id-prof = E.id-prof \wedge M.id-mat = E.id-mat \wedge X.nume = P.nume \wedge X.prenume = P.prenume \wedge X.denumirea = M.denumirea \wedge X.nota = E.nota) \}$

**CRD:**  $\{ \langle w, x, y, z \rangle \mid \langle A, B, C, D, E \rangle \in Profesor \wedge \langle F, G, H \rangle \in Materie \wedge \exists \langle I, J, K, L, M \rangle \in Examen (w=B \wedge x=C \wedge y=G \wedge z=M \wedge i=14 \wedge A=J \wedge F=K) \}$

**Alteore:**



c) Găsiți nr-matr pt. studenții care au susținut un singur examen.

**SOL:**

```
SELECT e.nr-matr
FROM Examen e
WHERE e.nr-matr NOT IN (SELECT e1.nr-matr
                        FROM Examen e1, Examen e2
                        WHERE e1.id-mat != e2.id-mat
                        and e1.nr-matr = e2.nr-matr)
```

**ARG:**

$$a := \rho_{E(nr-matr, id-prof, id-mat, data, nota)}$$

$$b := \rho_{E_1(nr-matr, id-prof, id-mat, data, nota)}$$

$$c := E \bowtie nr-matr_1 = nr-matr_2 \ E_1$$

$$d := \sigma_{id-mat_1 \neq id-mat_2} (c)$$

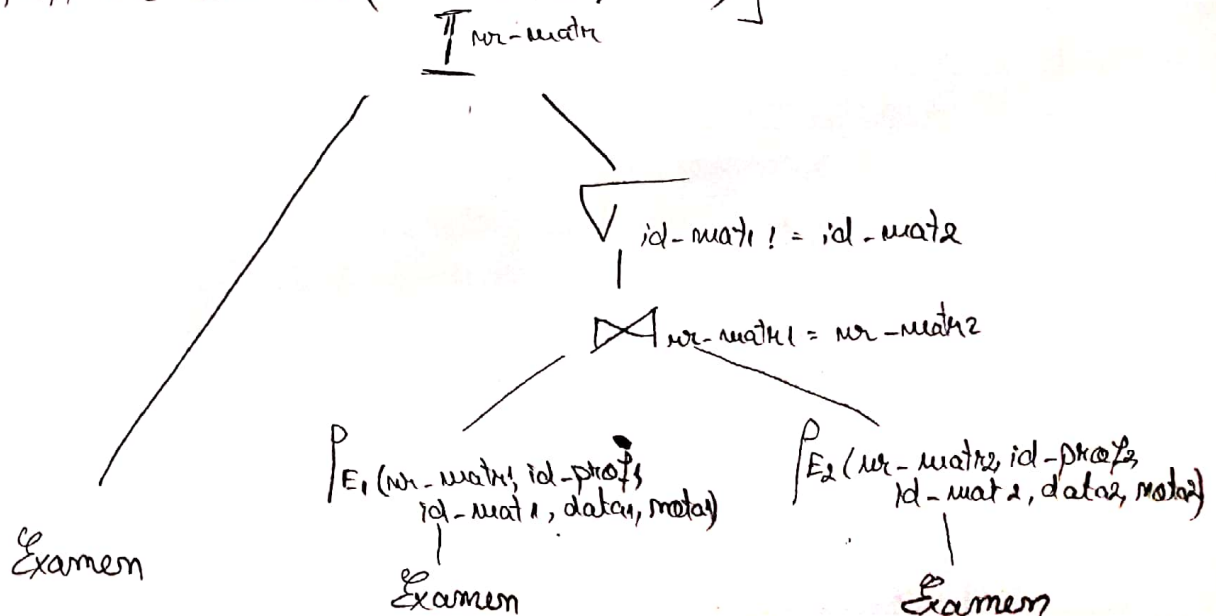
$$e := Examen - d$$

$$rez := \pi_{nr-matr} (e)$$

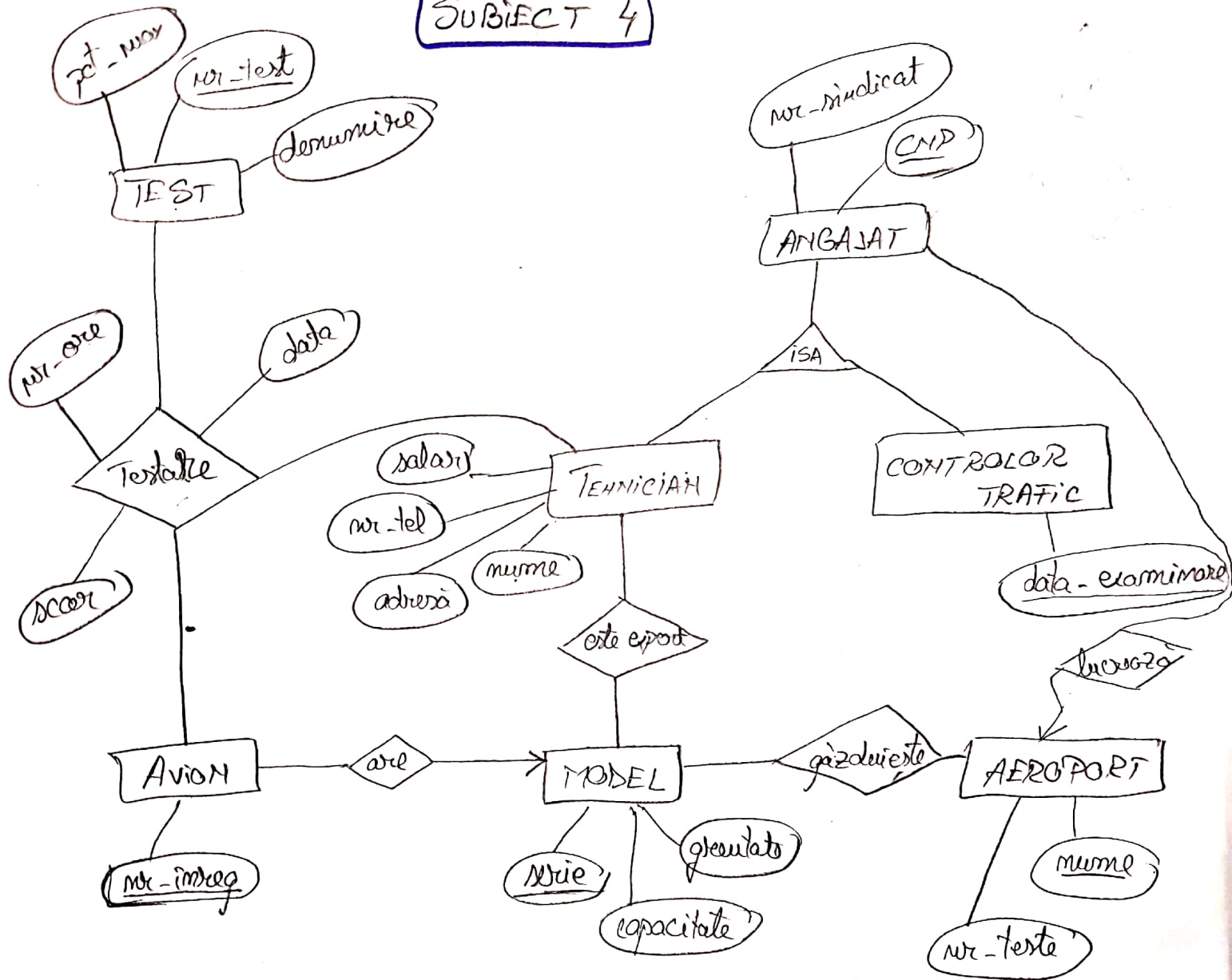
**CRT:**  $\{ E.nr-matr \mid E \in Examen \wedge \nexists (E_1 \in Examen \wedge E_2 \in Examen (E_1.nr-matr = E_2.nr-matr \wedge E_1.id-mat \neq E_2.id-mat)) \}$

**CRD:**  $\{ \langle A \rangle \mid \langle A, B, C, D, E \rangle \in Examen \wedge \nexists (\langle F, G, H, I, J, K \rangle \in Examen \wedge \langle L, M, N, O, P \rangle \in Examen (F=L \wedge H \neq N)) \}$

**Answer:**



# SUBJECT 4



XML

Sub. 4

<?xml version="1.0" standalone="yes"?>

<Acroport>

<nume> Andrei Iancu </nume>

<nr-texte> 200 </nr-texte>

<angajat>

<CNP> 26720040015 </CNP>

<nr-indicatii> 8 </nr-indicatii>

<tehnician>

<nume> Vasilescu Andrei </nume>

<adresa> Zorilor, 38 </adr.>

<nr-tel> 07.. </nr-tel>

<salari> 1200 </salari>

</tehnician>

<tehnician>

<nume> Pop Ion </nume>

<adr> ... </adr>

<nr-tel> 07.. </nr-tel>

<salari> 13000 </salari>

</tehnician>

<controlor-traffic>

<data-exam.> 07.05.2017 </data-ex.>

</controlor-traffic>

<controlor-traffic>

<data-ex.> 09.12.2018 </data-ex.>

</controlor-traffic>

</angajat>

<avion>

<nr-înreg> 0011 </nr-înreg>

<model>

<serie> ... </serie>

<capac.> ... </capac.>



```

<graduate> ... </graduate>
<technician>
...
</technician>
</model>
<model> ... </model>
<text>
  <pet-max> ... </pet-max>
  <nr-text> ... </nr-text>
  <denumire> ... </denumire>
</text>
<text> ... </text>
</arcam>
<arcam> ... </arcam>
</aeropart>

```

Sub. 4

XQUERY.

c) Sa' se da expresia XQUERY pt. a ofișă aeriană  
"Boeing 737" testată de tehnicianul "Vasilescu Andrei".

```
let $a = document("Aerospport.xml") / Aerospport
for $angajat in $a / angajat
  for $tehnician in $angajat / tehnician
    for $model in $tehnician / model
      where $tehnician / @nume =
        = "Vasilescu Andrei" and
        $model / @serie = "Boeing 737"
return <ret> { $aerian / @nr - inreg }
</ret>
```