**PROIECT BAZE DE DATE**

**Proiectarea unei baze de date care să gestioneze datele într-o sucursală bancară**

### 

**Matei Bianca Andreea**

Pentru realizarea acestui proiect am proiectat tabele care să definească următoarele entități:

* **ADRESĂ**
* **CLIENȚI**
* **CONTURI**
* **OPERAȚIUNI**
* **TRANZACȚII (tabel de legatură între operațiuni și conturi)**
* **ANGAJAȚI**

|  |  |
| --- | --- |
| **CLIENȚI** | **Constrângeri** |
| id\_client | primary key |
| nume |  |
| prenume |  |
| CNP/CUI | unic |
| tip\_client | check persoana fizica/persoana juridica |
| cod\_client | Unic (Ex: 10056734) |
| reprezentant\_bancar | foreign key pentru angajati(id\_angajat) |
| id\_adresa | foreign key pentru adresa(id\_adresa) |

Am urmărit ca tabele sa aibe următoarea structură:

|  |  |
| --- | --- |
| **ADRESA** | **Constrângeri** |
| id\_adresa | Primary key |
| țara |  |
| oraș |  |
| stradă |  |
| număr |  |
| bloc/clădire |  |

|  |  |
| --- | --- |
| **CONTURI** | **Constrângeri** |
| id\_cont | primary key |
| număr\_cont | unic (Ex: 20056375) |
| valoare |  |
| valută | check EUR/RON/USD |
| id\_client | foreign key pentru clienti(id\_client) |

|  |  |
| --- | --- |
| **OPERAȚIUNI** | **Constrângeri** |
| cod\_operațiune | primary key |
| tip\_operațiune | check DEPUNERE/EXTRAGERE/INFORMARE/ERROR |
| denumire |  |

|  |  |
| --- | --- |
| **Tranzacții** | **Constrângeri** |
| cont\_sursa | foreign key cont(id\_cont) |
| cont\_destinație | foreign key cont(id\_cont) |
| id\_operațiune | foreign key operatiuni(cod\_operatiune) |
| data\_tranzacție |  |
| valoare |  |
| detalii\_tranzacție |  |

|  |  |
| --- | --- |
| **Angajați** | **Constrângeri** |
| id\_angajat | primary key |
| nume |  |
| prenume |  |
| data\_nașterii |  |
| data\_angajării |  |
| CNP | unic |
| salariu |  |
| funcție/job |  |
| manager | foreign key angajati(id\_angajat) |
| id\_adresă | foreign key pentru adresa(id\_adresa) |

Am urmărit ca:

1. Între **adresă** și **clienți** să existe o legătură de 1:n, realizată prin definirea unui foreign key pe coloana **clienti.id\_adresa**;
2. Între **angajați** și **clienți** să existe o legătură de 1:n, realizată prin definirea unui foreign key pe coloana **clienti.reprezentant\_bancar**;
3. Între **clienti** și **conturi** există o legatură de 1:n, realizată prin definirea unui foreign key pe coloana **conturi.id\_client**;
4. Între **conturi** și **operațiuni** să existe o legătură de n:m, realizată prin definirea tabelului de legatură tranzacții și definirea foreign key-urilor pe coloanele **tranzactii.cont\_sursa**, **tranzactii.cont\_destinatie, tranzactii.id\_operatiune**;
5. Între **adresa** și **angajați** să existe o legatură de 1:n, realizată prin definirea unui foreign key pe coloana **angajati.id\_adresa**;
6. Între **angajați** și **angajați** să existe o legatură de 1:n, realizată prin definirea unui foreign key pe coloana **angajati.manager**;

Observații: Tabelul **tranzacții** conține detalii despre toate tranzacțiile realizate. Coloana valoare din acest tabel va fi întotdeauna reprezentată în RON. Tranzacțiile acceptate sunt cele din tabelul **operațiuni** și sunt definite mai jos. De exemplu, în cazul deschiderii unui cont sau adăugării unui cont nou, coloanele cont\_sursa și cont\_destinație va avea aceeași valoare, anume id-ul contului. În cazul unei tranzacții de transfer, coloana contul\_sursa conține id-ul contului de unde se va realiza transferul iar, cont\_destinație conține id-ul contului către care se va face transferul.

Am ales să rezolv următoarele cerințe:

1. Creați diagrama ERD al bazei de date mai sus mentionațe.
2. Creați baza de date și tabelele, respectând structura definită mai sus.
3. Modificați stuctura tabelului clienți, prin care să setați ca valoare DEFAULT pe coloana tip\_client, “PERSOANA FIZICĂ”;
4. Modificați stuctura tabelului clienți, prin care să ștergeți constrângerea de integritate referențiala dintre clienți și adresă (ștergeți foreign key-ul definit pe id\_adresă);
5. Modificați stuctura tabelului clienți, prin care să ștergeți coloana id\_adresă;
6. Modificați stuctura tabelului clienți, prin care să adaugați coloana id\_adresă;
7. Modificați stuctura tabelului clienți, prin care să adăugați o constrângere de integritate referențială între clienți și adresă (redefiniți foreign key-ul șters mai sus);
8. Rulați următoarele sintaxe insert:
   1. *USE banca;*
   2. *INSERT INTO operatiuni (cod\_operatiune, tip\_operatiune, denumire) VALUES*
   3. *(100, 'INFORMARE', 'Deschidere cont'),*
   4. *(101, 'INFORMARE', 'Adaugare cont'),*
   5. *(102, 'INFORMARE', 'Inchidere cont'),*
   6. *(200, 'INFORMARE', 'Transfer intre conturi'),*
   7. *(201, 'INFORMARE', 'Balanta cont'),*
   8. *(202, 'INFORMARE', 'Istoric tranzactii'),*
   9. *(300, 'DEPUNERE', 'Depunere ghiseu'),*
   10. *(301, 'DEPUNERE', 'Depunere ATM'),*
   11. *(400, 'EXTRAGERE', 'Extragere ghiseu'),*
   12. *(401, 'EXTRAGERE', 'Extragere ATM'),*
   13. *(500, 'ERROR', 'Fonduri insuficiente');*
9. Introduceți, folosind diferite sintaxe de insert, date în restul tabelelor (cel puțin 5-10 în fiecare).
10. Stergeți din tabela client, toți clienții care nu au conturi.
11. Modificați în RON valoarea din tabelul cont, al persoanelor ce au valuta în EUR, ținând cont de rata de schimb 1EUR=4.92RON.
12. Măriti cu 20% valoarea din tabela conturi a persoanelor juridice.
13. Afișați adresele din București.
14. Afișați conturile în RON a căror valoare este între 1000 și 2000.
15. Afișați conturile deschise în RON și EUR.
16. Afișați angajații al căror nume se termină în “escu”.
17. Afișați angajații cu id-urile 6,2,8,9,10.
18. Afișați persoanele fizice sub formă “nume prenume”, “CNP/CUI” , “ cod\_client”.
19. Afișați valoarea medie rotunjită la două zecimale a conturilor deschise în EUR, a celor deschise în RON și a celor deschise în USD.
20. Afișați suma tranzacționată de fiecare client în parte în lunile de vară, unde clientul a fost destinatar.;
21. Afișați suma tranzacționată de fiecare client în parte, pentru tranzacțiile cu id-ul 200, 202, 203, acolo unde clientul a fost sursa. Să se afișeze: „nume prenume” clientului, suma ieșită din cont sub forma „ < valoarea tranzacționată > RON”
22. Afișați valoarea tranzacționată în luna curentă.
23. Afișați numărul conturilor deschise în euro.
24. Afișați clienții cu conturi deschise. Afișare: "nume prenume", CNP/CUI, tip\_client, numar\_cont, valoare, valută;
25. Afișați clienții cu conturi deschise în RON și sediul lor. Afișare: "nume prenume", "țară, oraș, stradă, număr, bloc/clădire" ca: adresa, tip\_client, numar\_cont, valoare, valuta;
26. Afisați tranzacțiile în ultimele trei luni, ca: "tip\_operatiune denumire" ca operațiune, data\_tranzacție, valoare, detalii\_tranzacție;
27. Afișați numele clienților, numele reprezentanțiilor bancari și operațiunile făcute.
28. Afișați: "nume prenume" nume\_client, "nume prenume" reprezentant\_bancar, "tip\_operatiune denumire" operatiune, detalii\_tranzactie
29. Afișați clienții, angajații și adresa lor, folosind subinterogari sub forma: "nume prenume" client, id\_adresa, 'Client' "nume prenume" angajat, id\_adresa, 'Angajat'
30. Afișați, folosind subinterogări, angajații care sunt manageri.

Afișare: nume, prenume, data\_angajării, CNP, salariu, funcție/job. Coloana valoare din tabelul tranzacții reprezintă valoarea în RON.

1. Creați un view cu clienții, angajații și adresa lor, sub forma:

"nume prenume" client, "tara, oras, strada, numar, bloc/cladire" adresa, 'Client'

1. Creați un view cu angajații care nu sunt manageri și de portofoliul a căror clienți se ocupă.

Afișare: nume, prenume, data\_angajării, CNP, salariu, funcție/job, nr\_clienți

1. Creați o tabelă temporară cu angajații care sunt manageri și cei care nu sunt manageri.

Afișare: "nume prenume" Angajat,"nume prenume" Manager, data\_angajării, CNP, salariu, funcție/job

1. Creați un view care să conțină un raport general cu tranzacțiile efectuate în luna curentă, în funcție de operațiunea efectuată.

Afișare: "tip\_operațiune denumire" operațiune, valoare\_totală, valoare\_medie rotunjită la două zecimale . Să se afișeze doar dacă valoarea\_totală depășește valoarea\_medie.

1. Creați un view cu detaliile: nume, prenume, cod\_client, număr\_cont, valoare (tabelă clienți și cont);
2. Folosind view-ul creat, inserați un rând nou în tabelul clienți;
3. Creați un index compus pe tabela clienți, pe coloanele id\_client, cod\_client;
4. Creați un index simplu pe tabela cont, coloana numar\_cont;
5. Creați un index unic pe tabela angajați, coloana id\_adresa.
6. Ștergeți indecșii creați anterior.

**Diagrama ERD:**



**Programul:**

CREATE DATABASE banca;

USE banca;

CREATE TABLE adresa (

id\_adresa INT PRIMARY KEY AUTO\_INCREMENT NOT NULL,

tara VARCHAR(45),

oras VARCHAR (45),

strada VARCHAR (45),

numar VARCHAR (45),

bloc VARCHAR (45)

);

CREATE TABLE angajati(

id\_angajat INT PRIMARY KEY AUTO\_INCREMENT NOT NULL,

nume VARCHAR (45),

prenume VARCHAR(45),

data\_nasterii DATE,

data\_angajarii DATE,

CNP VARCHAR(45) UNIQUE,

salariu VARCHAR(45),

functie VARCHAR(45),

manager INT NOT NULL,

id\_adresa INT

);

CREATE TABLE clienti (

id\_client INT PRIMARY KEY,

nume VARCHAR(45),

prenume VARCHAR(45),

CNP\_CUI VARCHAR(45) UNIQUE,

tip\_client VARCHAR(45),

cod\_client VARCHAR (45) UNIQUE,

reprezentant\_bancar VARCHAR(45),

id\_adresa INT,

id\_angajat INT

);

CREATE TABLE conturi(

id\_cont INT PRIMARY KEY,

numar\_cont VARCHAR (45) UNIQUE,

valoare VARCHAR (45),

valuta VARCHAR (45),

id\_client INT

);

CREATE TABLE operatiuni(

cod\_operatiune INT PRIMARY KEY,

tip\_operatiune VARCHAR (45),

denumire VARCHAR (45),

id\_operatiune INT

);

CREATE TABLE tranzactii(

cont\_sursa VARCHAR(45),

cont\_destinatie VARCHAR(45),

id\_operatiune INT PRIMARY KEY AUTO\_INCREMENT NOT NULL ,

data\_tranzactie DATE,

valoare VARCHAR(45),

detalii\_tranzactie VARCHAR(45),

id\_cont INT

);

ALTER TABLE clienti ADD CONSTRAINT check\_tipclient

CHECK (tip\_client LIKE '%Persoana %Fizica' OR tip\_client LIKE '%Persoana %Juridica');

ALTER TABLE conturi ADD CONSTRAINT check\_valuta

CHECK (valuta LIKE '%EUR%' OR valuta LIKE '%RON%' OR valuta LIKE '%USD%' or valuta LIKE '%-%');

ALTER TABLE operatiuni ADD CONSTRAINT check\_tipoperatiune

CHECK (tip\_operatiune LIKE '%DEPUNERE%' OR tip\_operatiune LIKE '%EXTRAGERE%' OR tip\_operatiune LIKE '%INFORMARE%' OR tip\_operatiune LIKE '%ERROR%');

ALTER TABLE angajati ADD CONSTRAINT fk\_manager FOREIGN KEY (manager)

REFERENCES angajati(id\_angajat);

ALTER TABLE angajati ADD CONSTRAINT fk\_adresa2 FOREIGN KEY (id\_adresa)

REFERENCES adresa(id\_adresa);

ALTER TABLE clienti ADD CONSTRAINT fk\_adresa FOREIGN KEY (id\_adresa)

REFERENCES adresa (id\_adresa);

ALTER TABLE clienti ADD CONSTRAINT fk\_angajat FOREIGN KEY (id\_angajat)

REFERENCES angajati(id\_angajat);

ALTER TABLE conturi ADD CONSTRAINT fk\_clienti FOREIGN KEY (id\_client)

REFERENCES clienti (id\_client);

ALTER TABLE tranzactii ADD CONSTRAINT fk\_sursa FOREIGN KEY (id\_cont)

REFERENCES conturi (id\_cont);

ALTER TABLE tranzactii ADD CONSTRAINT fk\_destinatie FOREIGN KEY (id\_cont)

REFERENCES conturi (id\_cont);

ALTER TABLE tranzactii ADD CONSTRAINT fk\_operatiune FOREIGN KEY (id\_operatiune)

REFERENCES operatiuni (cod\_operatiune);

ALTER TABLE clienti CHANGE tip\_client tip\_client VARCHAR(45) DEFAULT 'PERSOANA FIZICA';

ALTER TABLE clienti DROP FOREIGN KEY fk\_adresa;

ALTER TABLE clienti DROP id\_adresa;

ALTER TABLE clienti ADD id\_adresa INT;

ALTER TABLE clienti ADD CONSTRAINT fk\_adresa FOREIGN KEY (id\_adresa)

REFERENCES adresa (id\_adresa);

USE banca;

INSERT INTO operatiuni (cod\_operatiune, tip\_operatiune, denumire) VALUES

(100, 'INFORMARE', 'Deschidere cont'),

(101, 'INFORMARE', 'Adaugare cont'),

(102, 'INFORMARE', 'Inchidere cont'),

(200, 'INFORMARE', 'Transfer intre conturi'),

(201, 'INFORMARE', 'Balanta cont'),

(202, 'INFORMARE', 'Istoric tranzactii'),

(300, 'DEPUNERE', 'Depunere ghiseu'),

(301, 'DEPUNERE', 'Depunere ATM'),

(400, 'EXTRAGERE', 'Extragere ghiseu'),

(401, 'EXTRAGERE', 'Extragere ATM'),

(500, 'ERROR', 'Fonduri insuficiente');

INSERT INTO adresa (id\_adresa,tara,oras,strada,numar,bloc) VALUES

(01, 'Romania', 'Bucuresti','Str. Constantin Disescu',18,3),

(02, 'Romania', 'Cluj','Str. Al.Ioan Cuza',20,10),

(03, 'Romania', 'Sibiu', 'Str. Frumoasa',15,7),

(04, 'Romania', 'Brasov','Str. Ion Creanga',16,3 ),

(05, 'Romania', 'Targoviste','Str. Mihai Eminescu',18,5),

(06, 'Romania', 'Bucuresti', 'Str. Iuliu Maniu',23,6),

(07, 'Romania', 'Bucuresti', 'Str. Iacob Negruzi',24,9),

(08, 'Romania', 'Bucuresti','Str. Ion Bianu',7,16),

(09, 'Romania', 'Iasi', 'Str. Stejar',18,4),

(010, 'Romania', 'Piatra-Neamt','Str. Alun',21,12),

(011, 'Romania', 'Craviova', 'Str. Ion Mincu',13,19);

INSERT INTO angajati (id\_angajat,nume,prenume,data\_nasterii,data\_angajarii,CNP,salariu,functie,manager,id\_adresa) VALUES

(001,'Ion','Maria','1972-09-08', '2000-11-05',2720908562347,4500,'Manager',001,01),

(002,'Popescu','Victor','1976-03-20', '2003-04-25',1760320896314,1700,'Ofiter Bancar',001,02),

(003,'Gheorghe','Alin','1980-07-20', '2006-08-09',180072056234,1700,'Ofiter Bancar',001,03),

(004,'Dumitru','Dan','1983-12-23', '2009-12-07',183122356234,1700,'Ofiter Bancar',001,04),

(005,'Ionescu','Paul','1989-09-20', '2008-08-10',189092056234,1700,'Ofiter Bancar',001,05),

(006,'Georgescu','Alexandru','1991-02-21', '2010-02-03',191022156234,1700,'Ofiter Bancar',001,06),

(007,'Teodorescu','Alina','1980-05-20', '2008-06-30',2800520623456,3500,'Manager',001,07),

(008,'Vicentiu','Paulina','1990-07-08', '2010-06-20',290070856234,1800,'Rprezentat Bancar',007,08),

(009,'Gogan','Elisabeta','1983-11-11', '2012-05-05',283111156234,1800,'Rprezentat Bancar',007,09),

(0010,'Panait','Victoria','1986-04-03', '2013-07-20',286040356234,1800,'Rprezentat Bancar',007,010),

(0011,'Alecsandrescu','Ioan','1992-02-08', '2015-03-10',192020856234,1800,'Rprezentat Bancar',007,011);

INSERT INTO clienti(id\_client,nume,prenume,CNP\_CUI,tip\_client,cod\_client,reprezentant\_bancar,id\_angajat,id\_adresa) VALUES

(1,'Vania','Ion','1967842310231','Persoana Fizica','1234','Gogan Elisabeta', '009','09'),

(2,'Dragos','Maria','2965478312654','Persoana Fizica','2345','Panait Victoria','0010','010'),

(3,'Enescu','Vlad','1234567894203', 'Persoana Fizica','3456', 'Panait Victoria','0010','010'),

(4,'Pavelescu','Florin','12356','Persoana Juridica','4567', 'Alecsandrescu Ioan','0011','011'),

(5,'Popescu','Victor','78653','Persoana Juridica','7891','Vicentiu Paulina','008','08'),

(6,'Pana','Alexandra','86545','Persoana Juridica','1685','Vicentiu Paulina','008','08'),

(7,'Petrescu','Victoria','12315','Persoana Juridica','3264','Vicentiu Paulina','008','08'),

(8,'Sanda','Maria','78564','Persoana Juridica','74523','Vicentiu Paulina','008','08');

INSERT INTO conturi(id\_cont,numar\_cont,valoare,valuta,id\_client) VALUES

(01,'RO02 INGB 0001 0082 1449 8950','2600','RON','1'),

(02,'RO57 BTRL RONC RT0V 0565 0501','3000','USD','2'),

(03,'RO28 BRDE 450S V210 0000 4500','8000', 'EUR','3'),

(04,'RO30 BRDE 450S V210 0000 4500','7500','RON','4'),

(05,'RO08 INGB 0001 0082 1450 8930','1800','RON','5');

INSERT INTO tranzactii(cont\_sursa, cont\_destinatie,id\_operatiune,data\_tranzactie,valoare,detalii\_tranzactie,id\_cont) VALUES

('RO02 INGB 0001 0082 1449 8950','RO09 INGB 0001 0082 1449 0730','200','2015-06-07','1500','Transfer intre conturi','01'),

('RO57 BTRL RONC RT0V 0565 0501','RO10 BTRL 0001 0082 1449 0730', '202', '2019-08-06','3000','Istoric Tranzactii',02),

('RO28 BRDE 450S V210 0000 4500', 'RO13 BRDE 0001 0082 1449 0730', '301','2018-06-08','5','Depunere Ghiseu',03),

('RO30 BRDE 450S V210 0000 4500', 'RO16 INGB 0001 0082 1449 0730', '500','2017-12-03','1200','Fonduri Insuficiente','04'),

('RO08 INGB 0001 0082 1450 8930','RO18 INGB 0001 0082 1450 8930','201','2019-05-23','600','Balanta','05');

**-- Stergeți din tabela client, toți clienții care nu au conturi.**

DELETE FROM conturi WHERE id\_client NOT IN (select id\_client from clienti);

**-- Modificați în RON valoarea din tabelul cont, al persoanelor ce au valuta în EUR, ținând cont de rata de schimb 1EUR=4.92RON.**

UPDATE conturi SET valoare=valoare\*4.92,valuta='RON'

WHERE valuta='EUR';

**-- Măriti cu 20% valoarea din tabela conturi a persoanelor juridice.**

SELECT valuta, avg(valoare) FROM conturi

WHERE valuta ='RON' OR valuta='EUR' OR valuta='USD'

GROUP BY valuta;

-- **Afișați adresele din București.**

SELECT \* FROM adresa

WHERE oras ='Bucuresti';

**-- Afișați conturile în RON a căror valoare este între 1000 și 2000.**

SELECT \* FROM conturi

WHERE valuta ='RON' AND valoare BETWEEN 1000 AND 2000;

**-- Afișați conturile deschise în RON și EUR.**

SELECT \* FROM conturi

WHERE valuta ='RON' OR valuta='EUR';

**-- Afișați angajații al căror nume se termină în “escu”.**

SELECT \* FROM angajati

WHERE right(nume,4)='escu';

**-- Afișați angajații cu id-urile 6,2,8,9,10.**

SELECT \* FROM angajati

WHERE id\_angajat IN (6,2,8,9,10);

**-- Afișați persoanele fizice sub formă “nume prenume”, “CNP/CUI” , “ cod\_client”.**

SELECT concat(nume,' ',prenume) Client,CNP\_CUI, cod\_client FROM clienti

WHERE tip\_client='Persoana Fizica';

**-- Afișați valoarea medie rotunjită la două zecimale a conturilor deschise în EUR, a celor deschise în RON și a celor deschise în USD.**

SELECT valuta, avg(valoare) FROM conturi

WHERE valuta ='RON' OR valuta='EUR' OR valuta='USD'

GROUP BY valuta; -- trebuie folosit double ?

**-- Afișați suma tranzacționată de fiecare client în parte în lunile de vară, unde clientul a fost destinatar.;**

SELECT sum(T.valoare) SumaTranzactionata, c.id\_client

FROM clienti C JOIN conturi CO using(id\_client)

JOIN tranzactii T ON T.cont\_destinatie=CO.numar\_cont

WHERE MONTH(t.data\_tranzactie) in (6, 7, 8)

GROUP BY c.id\_client;

**-- Afișsați suma tranzacționată de fiecare client în parte, pentru tranzacțiile cu id-ul 200, 202, 203, acolo unde clientul a fost sursa. Să se afișeze: „nume prenume” clientului, suma ieșită din cont sub forma „ < valoarea tranzacționată > RON”**

SELECT sum(T.valoare) SumaTranzactionata, T.id\_operatiune,concat\_ws(' ',nume,prenume,T.valoare,valuta) ValoareTranzactionata

FROM clienti JOIN conturi CO using(id\_client)

JOIN tranzactii T ON T.cont\_sursa=CO.numar\_cont

WHERE valuta='RON' AND T.id\_operatiune IN (200,202,203)

GROUP BY id\_client;

**-- Afișați valoarea tranzacționată în luna curentă.**

SELECT sum(valoare) FROM tranzactii

WHERE MONTH(data\_tranzactie)= month(current\_date());

**-- Afișați numărul conturilor deschise în euro.**

SELECT count(valuta) FROM conturi

WHERE valuta='EUR';

**-- Afișați clienții cu conturi deschise. Afișare: "nume prenume", CNP/CUI, tip\_client, numar\_cont, valoare, valută;**

SELECT concat(C.nume, ' ', C.prenume) Client,C.CNP\_CUI,C.tip\_client,CO.numar\_cont,CO.valoare,CO.valuta

FROM clienti C JOIN conturi CO

ON C.id\_client=CO.id\_client;

**-- Afișați clienții cu conturi deschise în RON și sediul lor. Afișare: "nume prenume", "țară, oraș, stradă, număr, bloc/clădire" ca: adresa, tip\_client, numar\_cont, valoare, valuta;**

SELECT

concat(C.nume, ' ', C.prenume) Client,concat(A.tara,' ',A.oras,' ',A.strada,' ',A.numar,' ',A.bloc) Adresa,C.tip\_client,

CO.numar\_cont,CO.valoare,CO.valuta

FROM clienti C JOIN conturi CO using(id\_client)

JOIN adresa A using(id\_adresa)

WHERE valuta='RON';

**-- Afisați tranzacțiile în ultimele trei luni, ca: "tip\_operatiune denumire" ca operațiune, data\_tranzacție, valoare, detalii\_tranzacție;**

SELECT T.data\_tranzactie,T.valoare,T.detalii\_tranzactie,concat(O.tip\_operatiune,' ',O.denumire) Operatiune

FROM tranzactii T JOIN operatiuni O

ON T.id\_operatiune=O.cod\_operatiune;

WHERE data\_tranzactie > current\_date() -90;

**-- Afișați numele clienților, numele reprezentanțiilor bancari și operațiunile făcute.**

**Afișați: "nume prenume" nume\_client, "nume prenume" reprezentant\_bancar, "tip\_operatiune denumire" operatiune, detalii\_tranzactie**

SELECT DISTINCT concat(C.nume,' ', C.prenume) nume\_client,concat(A.nume,' ',A.prenume)reprezentant\_bancar,

concat(O.tip\_operatiune,' ',O.denumire) operatiune,T.detalii\_tranzactie

FROM clienti C JOIN angajati A ON

C.id\_angajat=A.id\_angajat

JOIN tranzactii T JOIN operatiuni O ON

T.id\_operatiune=O.cod\_operatiune;

**-- Afișați clienții, angajații și adresa lor, folosind subinterogari sub forma:**

**"nume prenume" client, id\_adresa, 'Client'**

**"nume prenume" angajat, id\_adresa, 'Angajat'**

SELECT concat(nume, ' ', prenume) Client,

(SELECT concat\_ws(' ', tara, oras, strada,numar,bloc) FROM adresa WHERE id\_adresa=cln.id\_adresa) AS Adresa ,'Client' Observatii

FROM clienti cln

UNION

SELECT concat(nume,' ',prenume) Angajat,

(SELECT concat\_ws(' ',nume,prenume,functie) FROM angajati WHERE id\_adresa=ang.id\_adresa) AS Angajat, 'Angajat' Observatii

FROM angajati ang;

**-- Afișați, folosind subinterogări, angajații care sunt manageri.**

**Afișare: nume, prenume, data\_angajării, CNP, salariu, funcție/job**

SELECT concat(nume,' ',prenume),

(SELECT concat\_ws(' ',nume,prenume) FROM angajati WHERE functie='Manager') AS Manager FROM angajati;

**-- Creați un view cu clienții, angajații și adresa lor, sub forma:**

**"nume prenume" client, "tara, oras, strada, numar, bloc/cladire" adresa, 'Client'**

**"nume prenume" angajat, "tara, oras, strada, numar, bloc/cladire" adresa, 'Angajat'**

CREATE VIEW pacpac AS

SELECT concat(nume, ' ', prenume) Client,

(SELECT concat\_ws(' ', tara, oras, strada,numar,bloc) FROM adresa WHERE id\_adresa=cln.id\_adresa) AS Adresa ,'Client' Observatii

FROM clienti cln

UNION

SELECT concat(nume,' ',prenume) Angajat,

(SELECT concat\_ws(' ',tara,oras,strada,numar,bloc) FROM adresa WHERE id\_adresa=ang.id\_adresa) AS Adresa, 'Angajat' Observatii

FROM angajati ang;

**-- Creați un view cu angajații care nu sunt manageri și de portofoliul a căror clienți se ocupă. Afișare: nume, prenume, data\_angajării, CNP, salariu, funcție/job, nr\_clienți**

CREATE VIEW pacpac1 AS

SELECT A.nume,A.prenume,A.data\_angajarii,A.CNP,A.salariu,A.functie, count(A.id\_angajat)

FROM angajati A JOIN clienti C ON A.id\_angajat=C.id\_angajat

WHERE functie!='Manager'

GROUP BY A.id\_angajat;

**-- Creați o tabelă temporară cu angajații care sunt manageri și cei care nu sunt manageri. Afișare: "nume prenume" Angajat,"nume prenume" Manager, data\_angajării, CNP, salariu, funcție/job**

CREATE VIEW pacpac2 AS

SELECT distinct concat\_ws(' ',A.nume,A.prenume,A.data\_angajarii,A.CNP,A.salariu,A.functie) ReprezentantBancar,

concat\_ws(' ',Ang.nume,Ang.prenume,Ang.data\_angajarii,Ang.CNP,Ang.salariu,Ang.functie) Manager

FROM angajati A, angajati Ang-- JOIN angajati Ang ON A.id\_angajat=Ang.id\_angajat

WHERE A.functie <> 'Manager' and a.manager = ang.id\_angajat and Ang.functie='Manager';

**-- Creați un view cu detaliile: nume, prenume, cod\_client, număr\_cont, valoare (tabelă clienți și cont); Folosind view-ul creat, inserați un rând nou în tabelul clienți;**

SELECT concat(O.tip\_operatiune,' ', O. denumire) Operatiune, T.valoare, avg(T.valoare)

FROM tranzactii T JOIN operatiuni O ON T.id\_operatiune=O.cod\_operatiune

HAVING T.valoare > avg(T.valoare) AND MONTH(current\_date)=MONTH(T.data\_tranzactie);

**-- Creați un index compus pe tabela clienți, pe coloanele id\_client, cod\_client;**

CREATE INDEX clienti ON clienti(id\_client,cod\_client);

**-- Creați un index simplu pe tabela cont, coloana numar\_cont;**

CREATE UNIQUE INDEX conturi ON conturi(numar\_cont);

**-- Creați un index unic pe tabela angajați, coloana id\_adresa.**

CREATE UNIQUE INDEX angajati ON angajati(id\_adresa);

**-- Ștergeți indecșii creați anterior.**

DROP INDEX clienti ON clienti;

DROP INDEX conturi ON conturi;

DROP INDEX angajati ON angajati;