**Proiect nr. 8**

Cioabă Armina-Mihaela

AN II, Grupa 2

Baze de date, 2020

Se considera o societate de creditare, pentru persoane fizice si juridice. Datele sunt structurate

în următoarele tabele:

- **clienti\_fizice** având următoarele câmpuri: ID, nume, prenume, telefon, CNP, adresa.

Tabela cuprinde atât clienții persoane fizice cât și reprezentanții firmelor care sunt

stocați în tabela **clienti\_juridice**.

- clienti\_juridice având următoarele câmpuri: ID, nume, telefon, CUI, adresa,

ID\_reprez. O firmă poate avea un singur reprezentant.

- **credite** având următoarele câmpuri: IDcredit, IDclient, tip\_client, suma, perioada,

comision, dobanda, achitat. Un client poate avea mai multe credite pe numele sau.

- **plati** având următoarele câmpuri: IDplata, IDcredit, data, suma. Pentru un credit se pot

face oricâte plăți.

- **garantii** având următoarele câmpuri: IDgarantie, IDcredit, descriere, suma, tip. Un

credit poate avea mai multe garanții. O garanție nu poate fi utilizata la mai multe

credite.

Cerințe:

1. Sa se scrie comenzile de creare a tabelelor, impunând toate constrângerile care sunt

considerate utile din punct de vedere funcțional și al integrității datelor.

2. Sa se scrie comenzile de populare a tabelelor cu informații (minim 5 articole in fiecare

tabela).

3. Sa se afișeze totalul de bani aflați pe piața, sub forma:

Clienti Plasament

Persoane fizice xxxxxxxxx

Persoane juridice xxxxxxxxx

Total xxxxxxxxx

4. Sa se afișeze resursele financiare disponibile ale societății (banii care pot fi oferiți spre

creditare clienților) știind că inițial societatea avea un capital de 1.000.000.

5. Știind că reprezentanții unei firme sunt stocați în tabela **clienti\_fizice**, indiferent daca

au credite pe numele lor sau nu, legătura fiind asigurata de coloana ID\_reprez, sa se

scrie interogarea care afișează o lista cu toți clienții persoane fizice care au credite atât

pe numele lor, cat si pe firmele a căror reprezentanți sunt.

6. Sa se scrie interogarea care afișează toate firmele care au aceiași reprezentanți, si au

credite, grupate pe reprezentanți. Afișarea va include: nume reprezentant, prenume

reprezentant, firma, CUI, suma creditata, suma achitata.

7. Știind că fiecare garanție este stocata in tabela garantii, și că un credit poate avea mai

multe garanții, sa se afișeze lista creditelor care au garanții, grupat pe numărul de

garanții.

8. Știind că tipul garanțiilor specificate in tabela **garantii** poate fi: mobil, imobil și cec,

plecând de la aceasta tabela sa se creeze o tabela **garant\_comb** având următoarele

câmpuri: IDcredit, tip\_garantie, suma.

Unde tip garanție este mobil, daca toate garanțiile creditului sunt mobile, imobil, daca

toate garanțiile creditului sunt imobile, cec, daca toate garanțiile creditului sunt cecuri si

respectiv combinat daca creditul are garanții de diferite tipuri.

9. Considerând tabela de la punctul 8 creata, să se scrie un trigger care la orice

modificare a tabelei garanții, actualizează automat și tabela garant\_com.

10. Sa se scrie un trigger care la orice modificare in tabela plati să actualizeze coloana

**achitat** în tabela credite.

Rezolvare proiect:

Înainte de a trece la rezolvarea explicită a cerințelor, se recomandă utilizarea comenzilor următoare, pentru a asigura faptul că nu există eventuale duplicate și pentru a păstra logica de funcționare intactă.

**DROP TABLE garantii;**

**DROP TABLE plati;**

**DROP TABLE credite;**

**DROP TABLE credite**

**DROP TABLE clienti\_juridice;**

**DROP TABLE clienti\_fizice;**

**DROP TRIGGER t1;**

**DROP TRIGGER t2;**

**DROP VIEW reprezentanti;**

**DROP VIEW vgarantii;**

1. Sa se scrie comenzile de creare a tabelelor, impunand toate constrangerile care sunt

considerate utile din punct de vedere functional si al integritatii datelor.

**SET AUTOCOMMIT ON;**

**SELECT SYSDATE FROM DUAL;**

**ALTER SESSION SET NLS\_DATE\_FORMAT=’DD.MM.YYYY’;**

**CREATE TABLE clienti\_fizice**

**(ID int PRIMARY KEY,**

**nume varchar(20) NOT NULL,**

**prenume varchar(20) NOT NULL,**

**telefon number(15) NOT NULL,**

**CNP char(13) NOT NULL,**

**adresa varchar(30) NOT NULL);**

**CREATE TABLE clienti\_juridice**

**(ID int PRIMARY KEY,**

**nume varchar(30) NOT NULL,**

**telefon varchar(15) NOT NULL,**

**CUI char(12) NOT NULL,**

**adresa varchar(50) NOT NULL,**

**ID\_reprez int REFERENCES clienti\_fizice(ID));**

**CREATE TABLE credite**

**(IDcredit int PRIMARY KEY,**

**IDclient int REFERENCES clienti\_fizice(ID) ON DELETE CASCADE,**

**tip\_client varchar(15) NOT NULL,**

**suma number(8,2),**

**perioada number(2) CHECK(perioada BETWEEN 1 AND 25),**

**comision number(4,2),**

**dobanda number(4,2),**

**achitat number(8,2));**

**CREATE TABLE plati**

**(IDplata int NOT NULL,**

**IDcredit int REFERENCES credite(IDcredit) ON DELETE CASCADE,**

**data date,**

**suma number(8,2) NOT NULL);**

**CREATE TABLE garantii**

**(IDgarantie int PRIMARY KEY,**

**IDcredit int REFERENCES credite(IDcredit) ON DELETE CASCADE,**

**descriere varchar(30),**

**suma number(8,2),**

**tip varchar(15) NOT NULL);**

2. Sa se scrie comenzile de populare a tabelelor cu informații (minim 5 articole in fiecare

tabela).

**INSERT INTO clienti\_fizice VALUES(1, 'Cioaba', 'Armina', 0736615135, 111, 'Bahna, Mehedinti, nr.64');**

**INSERT INTO clienti\_fizice VALUES(2, 'Bosoanca', 'Daniela', 0716615131, 222, 'Orsova, str. Bradului, nr. 3');**

**INSERT INTO clienti\_fizice VALUES(3, 'Muntean', 'Andreea', 0712215131, 333, 'Eselnita, Mehedinti, nr. 11');**

**INSERT INTO clienti\_fizice VALUES(4, 'Patrinjei', 'Robert', 7852688994, 444, 'Londra, UK, nr. 9');**

**INSERT INTO clienti\_fizice VALUES(5, 'Artinescu', 'Adrian', 0766551234, 555, 'Eselnita, Mehedinti, nr. 2');**

**INSERT INTO clienti\_juridice VALUES(100, 'Continental', 0755678876,'RO2217098', 'Timisoara, nr.77', 1);**

**INSERT INTO clienti\_juridice VALUES(101, 'Hella', 0755670076,'RO2996889812', 'Timisoara, nr. 5', 5);**

**INSERT INTO clienti\_juridice VALUES(102, 'Libertatea', 0753370076,'RO012345', 'Bucuresti, nr. 99', 4);**

**INSERT INTO clienti\_juridice VALUES(103, 'Romsilva', 0753371176,'RO9988776655', 'Bucuresti, nr. 73', 1);**

**INSERT INTO clienti\_juridice VALUES(104, 'Enel', 0753372276,'RO9876543', 'Craiova, nr. 22', 4);**

**INSERT INTO credite VALUES(10,1,'juridica', 90000, 3, 0.5, 2.2, 49000);**

**INSERT INTO credite VALUES(11,1,'fizica', 5000, 2, 0.5, 1.5, 1000);**

**INSERT INTO credite VALUES(12,1,'juridica', 100000, 10, 0.9, 1.8, 5000);**

**INSERT INTO credite VALUES(13,2,'fizica', 10000, 5, 0.7, 2.2, 5000);**

**INSERT INTO credite VALUES(14,3,'fizica', 9000, 7, 0.7, 1.9, 5755.50);**

**INSERT INTO credite VALUES(15,4,'juridica', 10000, 5, 0.5, 1.5, 2500.99);**

**INSERT INTO credite VALUES(16,4,'juridica', 10000, 3, 0.5, 2.2, 1000);**

**INSERT INTO credite VALUES(17,5,'juridica', 10000, 3, 0.5, 2.2, 1000);**

**INSERT INTO credite VALUES(18,5,'fizica', 9000, 3, 0.5, 2.2, 1000);**

**INSERT INTO plati VALUES(20, 16, '11.05.2020', 1000);**

**INSERT INTO plati VALUES(21, 11, '09.05.2020', 1000);**

**INSERT INTO plati VALUES(22, 11, '09.05.2020', 5000);**

**INSERT INTO plati VALUES(23, 17, '05.05.2020', 500);**

**INSERT INTO plati VALUES(24, 12, '05.05.2020', 2500);**

**INSERT INTO plati VALUES(25, 12, '03.05.2020', 2500);**

**INSERT INTO plati VALUES(26, 13, '03.05.2020', 2500);**

**INSERT INTO plati VALUES(27, 14, '03.05.2020', 755.50);**

**INSERT INTO plati VALUES(28, 14, '01.05.2020', 5000);**

**INSERT INTO garantii VALUES(30,11,'executare silita', 10000,'imobil');**

**INSERT INTO garantii VALUES(31,12, 'executare silita', 10000,'imobil');**

**INSERT INTO garantii VALUES(32,13, 'executare silita', 1000,'mobil');**

**INSERT INTO garantii VALUES(33,14,'-', 900,'cec');**

**INSERT INTO garantii VALUES(34,15, 'executare silita', 1000,'imobil');**

**INSERT INTO garantii VALUES(35,15, 'executare silita', 500,'imobil');**

**INSERT INTO garantii VALUES(36,11, 'executare silita', 500,'mobil');**

3. Sa se afișeze totalul de bani aflați pe piața, sub forma:

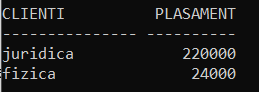
Clienti Plasament

Persoane fizice xxxxxxxxx

Persoane juridice xxxxxxxxx

Total xxxxxxxxx

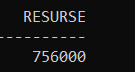
**SELECT tip\_client AS Clienti, sum(suma) AS Plasament FROM credite GROUP BY tip\_client;**



4. Sa se afișeze resursele financiare disponibile ale societății (banii care pot fi oferiți spre

creditare clienților) știind că inițial societatea avea un capital de 1.000.000.

**SELECT 1000000-sum(suma) AS resurse FROM credite;**



5. Știind că reprezentanții unei firme sunt stocați în tabela **clienti\_fizice**, indiferent daca

au credite pe numele lor sau nu, legătura fiind asigurata de coloana ID\_reprez, să se

scrie interogarea care afișează o lista cu toți clienții persoane fizice care au credite atât

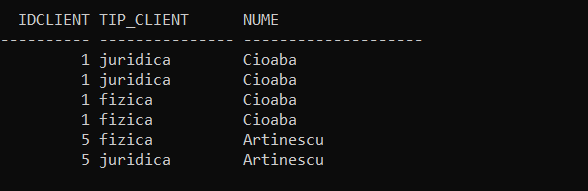
pe numele lor, cat si pe firmele a căror reprezentanți sunt.

**CREATE VIEW reprezentanti AS (SELECT \* FROM credite);**

**SELECT credite.IDclient, credite.tip\_client,nume**

**FROM credite, reprezentanti,clienti\_fizice**

**WHERE reprezentanti.tip\_client<>credite.tip\_client AND clienti\_fizice.ID=reprezentanti.IDclient AND credite.IDclient=clienti\_fizice.ID;**



6. Sa se scrie interogarea care afișează toate firmele care au aceiași reprezentanți, și au

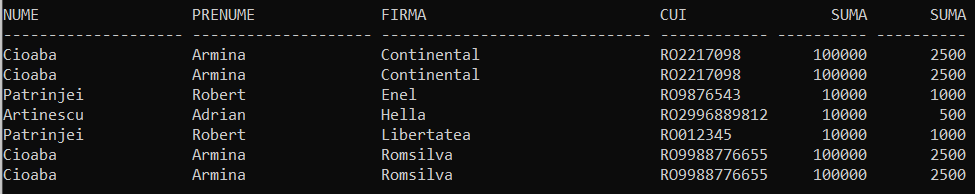
credite, grupate pe reprezentanți. Afișarea va include: nume reprezentant, prenume

reprezentant, firma, CUI, suma creditata, suma achitata.

**SELECT clienti\_fizice.nume, clienti\_fizice.prenume, clienti\_juridice.nume AS firma, clienti\_juridice.CUI, credite.suma, plati.suma**

**FROM clienti\_fizice, clienti\_juridice, credite, plati**

**WHERE clienti\_fizice.ID=clienti\_juridice.ID\_reprez AND clienti\_fizice.ID=credite.IDclient AND credite.IDcredit=plati.IDcredit AND tip\_client like('juridica') ORDER BY clienti\_juridice.nume;**



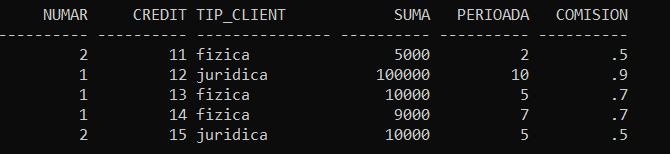
7. Știind că fiecare garanție este stocata in tabela garantii, și că un credit poate avea mai

multe garanții, să se afișeze lista creditelor care au garanții, grupat pe numărul de

garanții.

**CREATE VIEW vgarantii AS (SELECT count(IDcredit) AS numar, IDcredit AS credit FROM garantii GROUP BY IDcredit);**

**SELECT numar, credit, tip\_client, suma, perioada, comision FROM vgarantii, credite WHERE credit=credite.IDcredit;**



8. Știind că tipul garanțiilor specificate in tabela **garantii** poate fi: mobil, imobil și cec,

plecând de la aceasta tabela sa se creeze o tabela **garant\_comb** având următoarele

câmpuri: IDcredit, tip\_garantie, suma.

Unde tip garanție este *mobil*, daca toate garanțiile creditului sunt mobile, *imobil*, daca

toate garanțiile creditului sunt imobile, *cec*, daca toate garanțiile creditului sunt cecuri si

respectiv *combinat* daca creditul are garanții de diferite tipuri.

**CREATE TABLE garant\_comb(IDcredit,suma) AS SELECT IDcredit,sum(suma) FROM garantii GROUP BY IDcredit;**

**ALTER TABLE garant\_comb ADD tip\_garantie varchar(8);**

**CREATE OR REPLACE FUNCTION fgarantii(ID int)**

**RETURN varchar**

**AS c\_mobil int:=0;**

**c\_imobil int:=0;**

**c\_cec int:=0;**

**aux varchar(10);**

**CURSOR c1 IS SELECT tip FROM garantii WHERE IDcredit=ID;**

**BEGIN**

**OPEN c1;**

**LOOP**

**FETCH c1 INTO aux;**

**EXIT WHEN c1%notfound;**

**IF aux='mobil' THEN c\_mobil:=c\_mobil+1;**

**END IF;**

**IF aux='imobil' THEN c\_imobil:=c\_imobil+1;**

**END IF;**

**IF aux='cec' THEN c\_cec:=c\_cec+1;**

**END IF;**

**END LOOP;**

**CLOSE c1;**

**IF(c\_mobil>0 AND c\_imobil=0 AND c\_cec=0) THEN**

**RETURN 'mobil';**

**ELSE IF (c\_mobil=0 AND c\_imobil>0 AND c\_cec=0) THEN**

**RETURN 'imobil';**

**ELSE IF (c\_mobil=0 AND c\_imobil=0 AND c\_cec>0) THEN**

**RETURN 'cec';**

**ELSE RETURN 'combinat';**

**END IF;**

**END IF;**

**END IF;**

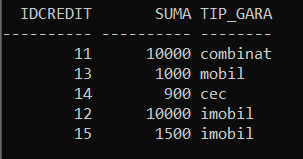
**END;**

**/**

**UPDATE garant\_comb SET tip\_garantie=fgarantii(IDcredit);**

**SELECT \* FROM garant\_comb;**

Se observă că procedura a fost executată cu succes și rezultatul este cel dorit. S-a utilizat cursorul c1 pentru a putea lucra cu mai multe linii simultan.



9. Considerând tabela de la punctul 8 creata, să se scrie un trigger care la orice

modificare a tabelei garanții, actualizează automat și tabela garant\_com.

**CREATE OR REPLACE TRIGGER t1**

**AFTER INSERT OR UPDATE ON garantii**

**FOR EACH ROW**

**BEGIN**

**UPDATE garant\_comb SET suma=suma+:new.suma WHERE IDcredit=:new.IDcredit;**

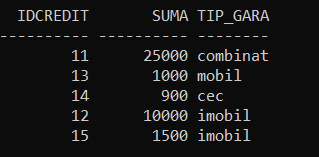
**END;**

**/**

**UPDATE garantii SET suma=15000 WHERE IDgarantie=30;**

**SELECT \* FROM garant\_comb;**

Se observă că trigger-ul s-a efectuat cu succes și tabela garant\_comb a fost actualizată când s-a făcut actualizarea tabelei garantii.



10. Sa se scrie un trigger care la orice modificare in tabela plati să actualizeze coloana

**achitat** în tabela credite.

**CREATE OR REPLACE TRIGGER t2**

**AFTER INSERT OR UPDATE ON plati**

**FOR EACH ROW**

**BEGIN**

**UPDATE credite SET achitat=achitat+:new.suma WHERE credite.IDcredit=:new.IDcredit;**

**END;**

**/**

**INSERT INTO plati VALUES(29, 11, '25.05.2020', 9000);**

**SELECT \* FROM credite WHERE IDcredit=11;**

Se observă că trigger-ul a actualizat tabela credite la o noua inserare în tabela plati.

