Baza de date a unei fabrici de geci

Realizat de: Suditu Darius-Cosmin

Cuprins

1. Descrierea modelului
2. Constrangeri (restrictii, reguli)
3. Entitati
4. Relatii
5. Atribute
6. Diagrama Entitate-Relatie
7. Diagrama Conceptuala
8. Schema Relationala
9. Normalizarea
   1. NON FN1 – FN1
   2. NON FN2 – FN2
   3. NON FN3 – FN3
10. Crearea unei secvente
11. Crearea tabelelor si inserarea datelor
12. Cereri SQL
    1. Cerinta1 + Rezolvare1
    2. Cerinta2 + Rezolvare2
    3. Cerinta3 + Rezolvare3
    4. Cerinta4 + Rezolvare4
    5. Cerinta5 + Rezolvare5
13. Actualizare si Suprimare

1. Descrierea modelului real, a utilității acestuia și a regulilor de funcționare.

Pentru acest proiect, am ales realizarea bazei de date a unei fabrici de geci, ce se ocupa de asemenea si cu depozitarea si comercializarea acestora. Exista diferite locatii pentru fiecare fabrica, depozit sau magazin fizic in diferite tari, angajati care lucreaza fie in fabrica fie in magazinul fizic. Clientii plaseaza comenzi si achizitoneaza astfel geci din magazine.

1. Prezentarea constrângerilor (restricții, reguli) impuse asupra modelului.

Restrictii si reguli:

* Fiecare angajat lucreaza intr-o singura locatie, iar fiecare locatie are cel putin un angajat.
* Angajatii lucreaza fie in magazine, fie in fabrici, avand diferite specializari.Acestia au un id unic de angajat dar si un id pentru departamentul din care fac parte.
* Fiecare comanda este unica, identifiabila prin ID.
* O geaca este produsa intr-o fabrica, apoi depozitata.
* Magazinele iau geci din depozit prin intermediul unui contract intern unic, identifiabil prin ID.
* Comenzile sunt unice si pot fi identificate printr-un id.
* Depozitele, fabricile si magazinele se afla in diverse locatii din diferite tari, identifiabile prin ID.

Constrangeri:

* Cheie primara – PK
* Cheie externa – FK

1. Descrierea entităților, incluzând precizarea cheii primare.

***CLIENT***(ID\_CLIENT#) – entitate ce va contine detalii despre un client care face o comanda.

***LOCATIE*** (ID\_LOCATIE#) – entitate ce va contine detalii despre locatiile in care se afla magazinele,depozitele,fabricile.

***MAGAZIN***(ID\_MAGAZIN#) – entitate ce va contine informatii despre un magazin din lantul de magazine.

***ANGAJAT*** (ID\_ANGAJAT#) – entitate ce contine detalii despre toti angajatii firmei.

***ANGAJAT\_MAGAZIN***(ID\_ANGAJAT\_MAGAZIN#) – entitate ce va contine detalii despre angajatii din magazinele fizice..

***FABRICA***(ID\_FABRICA#) – entitate ce va contine detalii despre o anumita fabrica din retea.

***MUNCITOR*** – entitate care contine detalii despre angajatii din fabrici.

***DEPOZIT***(ID\_DEPOZIT#) – entitate ce va contine detalii despre depozitele din retea.

***CONTRACT***(ID\_CONTRACT#) -entitate ce va contine detalii despre contractele realizate intre depozite si magazine fizice.

***GEACA*** (ID\_GEACA#) – entitate ce va contine detalii despre gecile fabricate.

***MATERIAL***(ID\_MATERIAL#) – entitate ce va contine detalii despre materialele folosite in confectionarea gecilor.

***CONSTRUIRE*** (ID\_GEACA#ID\_MATERIAL#) – entitate ce va contine detalii despre modul de confectionare al fiecarei geci.

***COMANDA***(ID\_COMANDA#) – entitate ce va contine detalii despre o comanda facuta de un client.

1. Descrierea relațiilor, incluzând precizarea cardinalității acestora.

***CLIENT-COMANDA:*** un client poate plasa mai multe comenzi, dar o comanda poate fi plasata de un singur client.

* Tipul relatiei: one-many
* Cardinalitate: 1-M(1)

***ANGAJATI-MUNCITOR,ANGAJAT\_MAGAZIN***: angajatul este fie muncitor fie angajat in magazin.

* Tipul relatiei: ISA
* Cardinalitate: 1-1(0)

***GEACA-MATERIAL***: o geaca este confectionata din mai multe materiale, iar mai multe materiale sunt folosite pt confectionarea unei geci.

* Tipul relatiei: many-many
* Cardinalitate: M(1)-M(1)

***MUNCITOR-FABRICA***: intr-o fabrica pot lucra mai multi muncitori, dar un muncitor poate lucra intr-o singura fabrica.

* Tipul relatiei: many-one
* Cardinalitate: M(1)-1

***FABRICA-LOCATIE***: o fabrica corespunde unei locatii.

* Tipul relatiei: one-one
* Cardinalitate: 1-1

***MAGAZIN-LOCATIE:*** magazinul corespunde unei locatii.

* Tipul relatiei: one-one
* Cardinalitate: 1-1

***DEPOZIT-LOCATIE***: un depozit corespunde unei locatii..

* Tipul relatiei: one-one
* Cardinalitate: 1-1

***FABRICA-GEACA***: o fabrica poate fabrica mai multe geci, dar o geaca este fabricata intr-o singura fabrica

* Tipul relatiei: one-many
* Cardinalitate: 1-M(1)

***DEPOZIT-GEACA***: intr-un depozit pot fi mai multe geci, dar o geaca poate fi intr-un singur depozit.

* Tipul relatiei: one-many
* Cardinalitate: 1-M(0)

***DEPOZIT-CONTRACT***: un depozit incheie mai multe contracte, dar un contract poate fi incheiat cu un singur depozit.

* Tipul relatiei: one-many
* Cardinalitate: 1-M(0)

***MAGAZIN-CONTRACT***: un magazin incheie mai multe contracte, dar un contract poate fi incheiat cu un singur magazin.

* Tipul relatiei: one-many
* Cardinalitate: 1-M(0)

***MAGAZIN-ANGAJAT\_MAGAZIN***: intr-un magazin pot lucra mai multi angajati, dar un angajat poate lucra intr-un singur magazin.

* Tipul relatiei: one-many
* Cardinalitate: 1-M(1)

***MAGAZIN-COMANDA***: un magazin poate primi mai multe comenzi, dar o comanda se poate da dintr-un singur magazin.

* Tipul relatiei: one-many
* Cardinalitate: 1-M(0)

***COMANDA-GEACA***: o comanda poate contine un singur tip de geaca, dar o geaca poate fi in mai multe comenzi.

* Tipul relatiei: one-many
* Cardinalitate: 1-M(1)

1. Descrierea atributelor, incluzând tipul de date și eventualele constrângeri, valori implicite, valori posibile ale atributelor.

***CLIENT***

ID\_CLIENT# int not null

NUME\_CLIENT varchar(50) not null

EMAIL\_CLIENT varchar(50)

ID\_CLIENT# => cheie primara de tip int, diferita de null, care identifica unic un client..

NUME\_CLIENT => atribut de tip sir de caractere, diferit de null, de lungime maxima 50, ce va contine numele intreg al clientului.

EMAIL\_CLIENT => atribut de tip sir de caractere, de lungime maxima 50, ce va contine adresa de email a clientului.

***LOCATIE***

ID\_LOCATIE# int not null

TARA varchar2(20) not null

ORAS varchar2(20) not null

STRADA varchar2(30)

COD\_POSTAL varchar2(10) not null

ID\_LOCATIE # => cheie primara de tip int, diferita de null, care identifica unic o locatie

TARA => atribut de tip sir de caractere diferit de null ce va contine numele tarii

ORAS => atribut de tip sir de caractere diferit de null ce va contine numele orasului

STRADA => atribut de tip sir de caractere ce va contine numele strazii

COD\_POSTAL => atribut de tip sir de caractere ce va contine codul postal (10 caractere).

***MAGAZIN***

ID\_MAGAZIN# int not null

NUME varchar2(20) not null

TELEFON varchar2(10)

ID\_LOCATIE FK not null

ID\_MAGAZIN # => cheie primara de tip int, diferita de null, care identifica unic un magazin

NUME => atribut de tip sir de caractere, diferit de null, de lungime maxima 20, ce va contine numele magazinului.

TELEFON => atribut de tip sir de caractere, de lungime maxima 10, ce va contine nr de telefon al magazinului.

FK ID\_LOCATIE => cheie externa ce va identifica id-ul de locatie al locatiei in care este amplasat magazinul.

***ANGAJAT***

ID\_ANGAJAT# int not null

Nume\_Angajat varchar2(50) not null

Salariu\_Angajat int not null

Data\_Angajat date not null

Telefon\_Angajat varchar2(10)

ID\_ANGAJAT # => cheie primara de tip int, diferita de null, care identifica unic un angajat

NUME\_ANGAJAT => atribut de tip sir de caractere, diferit de null, de lungime maxima 50, ce va contine numele angajatului.

DATA\_ANGAJAT => atribut de tip sir data care va contine data angajarii fiecarui angajat.

TELEFON => atribut de tip sir de caractere, de lungime maxima 10, ce va contine nr de telefon al angajatului.

***ANGAJAT\_MAGAZIN***

ID\_ANGAJAT\_MAGAZIN# int not null

SPECIALIZARE varchar2(20) not null

ID\_ANGAJAT FK not null

ID\_MAGAZIN FK not null

ID\_ANGAJAT\_MAGAZIN# => cheie primara de tip int, diferita de null, care identifica unic un angajat din magazin

SPECIALIZARE => atribut de tip sir de caractere ce reprezinta specializarea angajatului

FK ID\_ANGAJAT => cheie externa ce va identifica id-ul de angajat al angajatului din magazin.

FK ID\_MAGAZIN => cheie externa ce va identifica id-ul de magazin al magazinului in care lucreaza angajatul.

***FABRICA***

ID\_FABRICA# int not null

PRODUCTIE int not null

DATA\_INFIINTARE date not null

ID\_LOCATIE FK not null

ID\_FABRICA # => cheie primara de tip int, diferita de null, care identifica unic o fabrica

PRODUCTIE => atribut de tip int, diferit de null, identifica cantitatea de productie a fabricii.

DATA\_INFIINTARE => atribut de tip date, diferit de null, care identifica data de infiintare a fabricii.

FK ID\_LOCATIE => cheie externa ce va identifica id-ul de locatie al locatiei in care este amplasat magazinul.

***MUNCITOR***

ID\_MUNCITOR# int not null

SPECIALIZARE varchar2(20) not null

ID\_ANGAJAT FK not null

ID\_FABRICA FK not null

ID\_MUNCITOR# => cheie primara de tip int, diferita de null, care identifica unic un muncitor din fabrica.

SPECIALIZARE => atribut de tip sir de caractere ce reprezinta specializarea muncitorului

FK ID\_ANGAJAT => cheie externa ce va identifica id-ul de angajat al angajatului din magazin.

FK ID\_FABRICA => cheie externa ce va identifica id-ul de fabrica al fabricii in care lucreaza

***DEPOZIT***

ID\_DEPOZIT# int not null

CAPACITATE int not null

ID\_LOCATIE FK not null

ID\_DEPOZIT # => cheie primara de tip int, diferita de null, care identifica unic un depozit

CAPACITATE => atribut de tip int, diferit de null, identifica capacitatea depozitului.

FK ID\_LOCATIE => cheie externa ce va identifica id-ul de locatie al locatiei in care este amplasat magazinul.

***CONTRACT***

ID\_CONTRACT# int not null

SUMA int not null

ID\_DEPOZIT FK not null

ID\_MAGAZIN FK not null

ID\_CONTRACT# => cheie primara de tip int, diferita de null, care identifica unic un contract incheiat

SUMA => atribut de tip int ce va identifica suma corespunzatoare contractului.

FK ID\_DEPOZIT => cheie externa ce va identifica id-ul depozitului ce incheie contractul cu magazinul.

FK ID\_MAGAZIN => cheie externa ce va identifica id-ul magazinului care incheie un contract cu depozitul.

***GEACA***

ID\_GEACA# int not null

NUME varchar2(20) not null

TIP varchar2(20) not null

MARIME varchar2(10) not null

ID\_FABRICA FK not null

ID\_DEPOZIT FK not null

ID\_GEACA# => cheie primara de tip int, diferita de null, care identifica unic o geaca fabricata.

NUME => atribut de tip sir de caractere ce reprezinta numele gecii.

TIP => atribut de tip sir de caractere ce reprezinta tipul gecii.

FK ID\_FABRICA => cheie externa ce va identifica id-ul de fabrica al fabricii in care este facuta geaca.

FK ID\_DEPOZIT => cheie externa ce va identifica id-ul de depozit al depozitului in care se afla geaca

***MATERIAL***

ID\_MATERIAL# int not null

NUME varchar2(50) not null

CANTITATE int

ID\_MATERIAL # => cheie primara de tip int, diferita de null, care identifica unic un material.

NUME => atribut de tip sir de caractere ce va contine numele unui material

CANTITATEA => atribut de tip int ce va spune cantitatea de material folosita.

***CONSTRUIRE***

PK FK ID\_GEACA

PK FK ID\_MATERIAL

***COMANDA***

ID\_COMANDA# int not null

CANTITATE int not null

PRET int not null

DATA\_COMANDA date

ID\_GEACA FK not null

ID\_MAGAZIN FK not null

ID\_CLIENT FK not null

ID\_COMANDA# => cheie primara de tip int, diferita de null, care identifica unic o comanda data.

CANTITATE => atribut de tip int ce va identifica numarul de geci comandate

PRET => atribut de tip int ce va identifica pretul total al unei comenzi

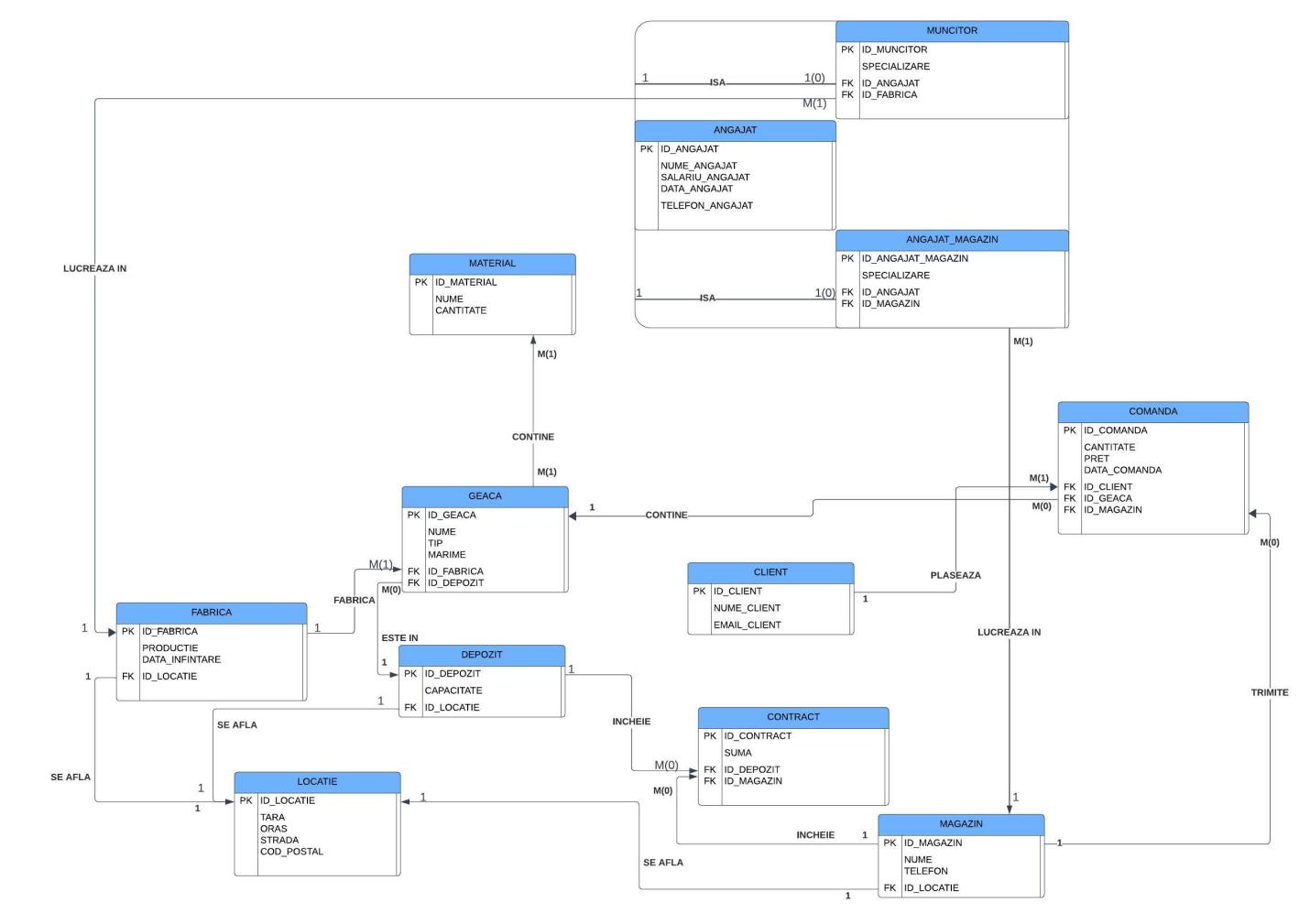
DATA\_COMANDA => atribut de tip date ce va identifica data in care a fost efectuata o comanda.

FK ID\_GEACA => cheie externa ce va identifica id-ul gecii comandate.

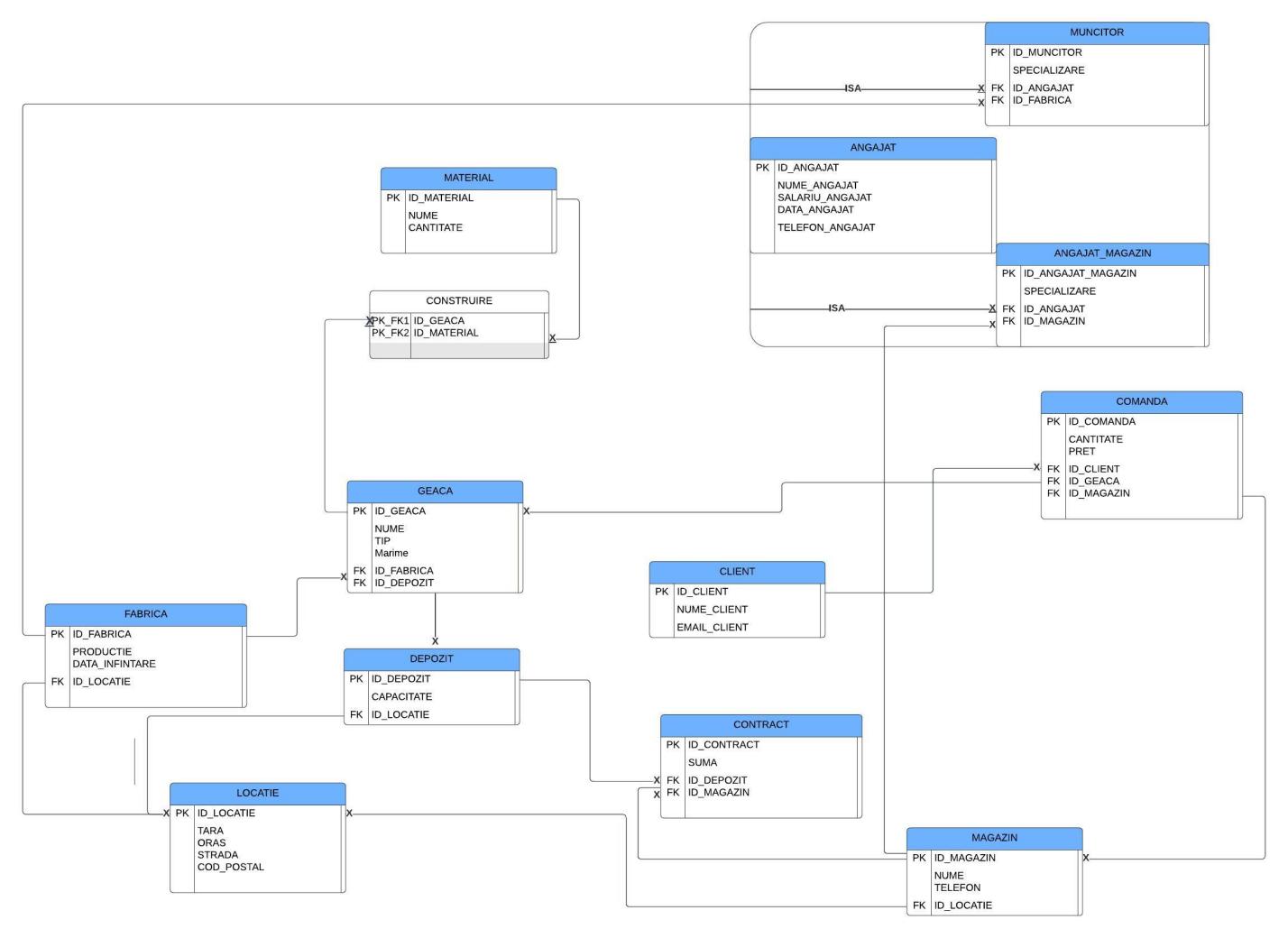
FK ID\_MAGAZIN => cheie externa ce va identifica id-ul magazinului din care este efectuata comanda

FK ID\_CLIENT => cheie externa ce va identifica id-ul clientului ce efectueaza comanda

1. Realizarea diagramei entitate-relație corespunzătoare descrierii de la punctele 3-5.



1. Realizarea diagramei conceptuale corespunzătoare diagramei entitate-relație proiectate la punctul 6. Diagrama conceptuală obținută trebuie să conțină minimum 6 tabele (fără considerarea subentităților), dintre care cel puțin un tabel asociativ.



1. Enumerarea schemelor relaționale corespunzătoare diagramei conceptuale proiectate la punctul 7.

***CLIENT***

(ID\_CLIENT#, NUME\_CLIENT, EMAIL\_CLIENT)

***LOCATIE***

(ID\_LOCATIE#, TARA, ORAS, STRADA, COD\_POSTAL)

***MAGAZIN***

(ID\_MAGAZIN#, NUME, TELEFON, ID\_LOCATIE FK)

***ANGAJAT***

(ID\_ANGAJAT#, NUME\_ANGAJAT, SALARIU\_ANGAJAT, DATA\_ANGAJT, TELEFON\_ANGAJAT)

***ANGAJAT\_,MAGAZIN***

(ID\_ANGAJAT\_MAGAZIN#, SPECIALIZARE, ID\_ANGAJAT FK)

***FABRICA***

(ID\_FABRICA#, PRODUCTIE, DATA\_INFIINTARE, ID\_LOCATIE FK)

***MUNCITOR***

(ID\_MUNCITOR#, SPECIALIZARE, ID\_ANGAJAT FK)

***DEPOZIT***

(ID\_DEPOZIT#, CAPACITATE, ID\_LOCATIE FK)

***CONTRACT***

(ID\_CONTRACT#, SUMA, ID\_DEPOZIT FK, ID\_MAGAZIN FK)

***GEACA***

(ID\_GEACA#, NUME, TIP, MARIME, ID\_FABRICA FK, ID\_DEPOZIT FK)

***MATERIAL***

(ID\_MATERIAL#, NUME)

***CONSTRUIRE***

(ID\_GEACA PK FK, ID\_MATERIAL PK FK)

***COMANDA***

(ID\_COMANDA#, CANTITATE, PRET, DATA\_COMANDA, ID\_GEACA FK, ID\_MAGAZIN FK, ID\_CLIENT FK)

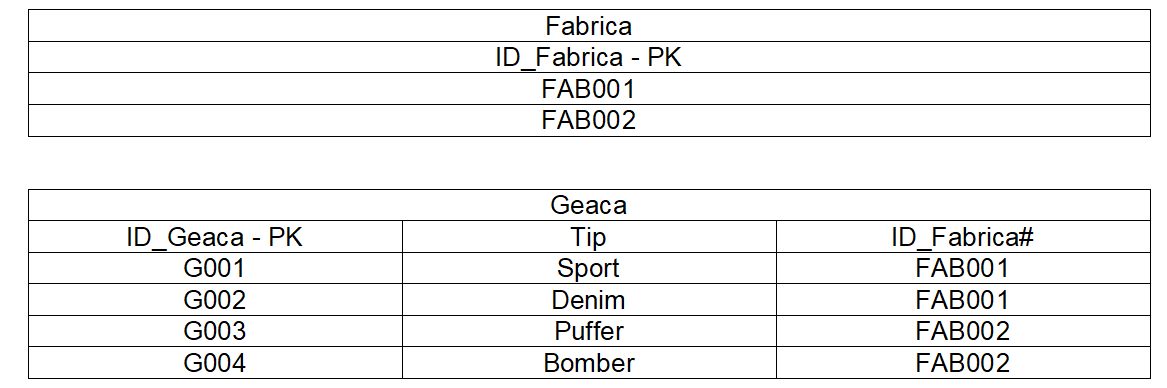
1. Realizarea normalizării până la forma normală 3 (FN1-FN3).

**NON FN1:**

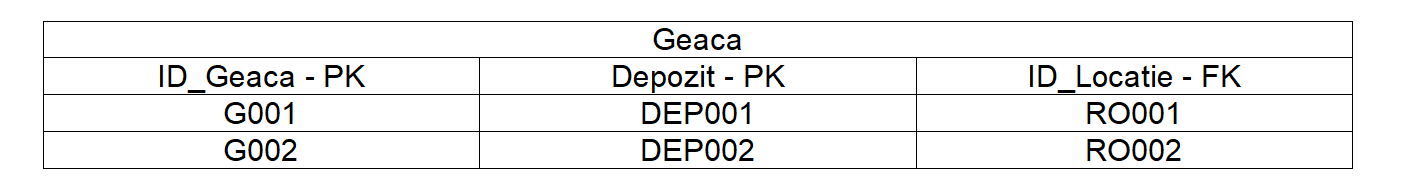


Realizam un nou tabel pentru geci, nu le adaugam in fabrica.

**FN1:**

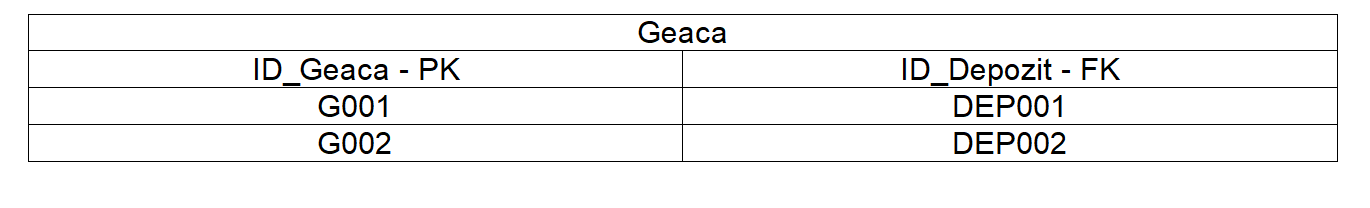


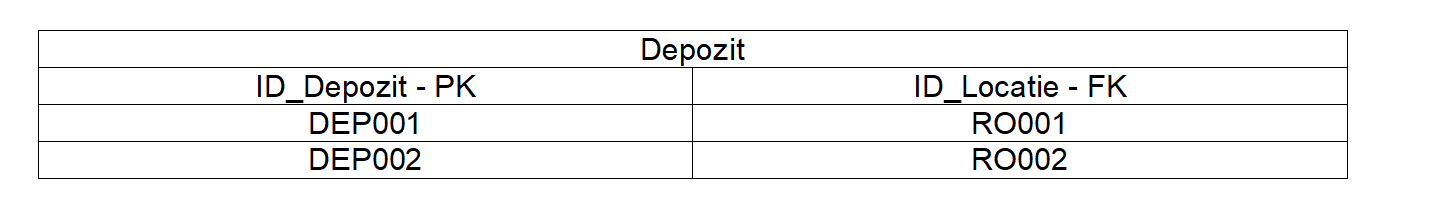
**NON FN2:**



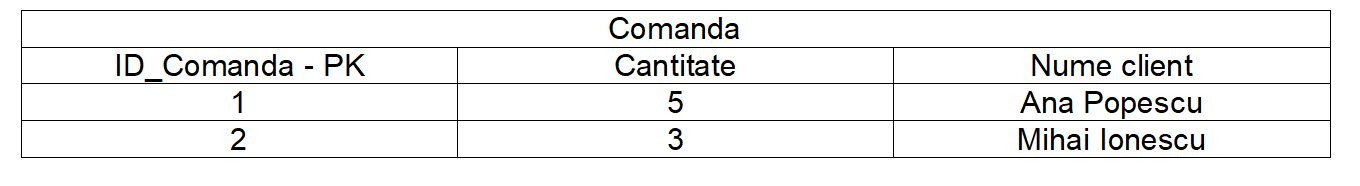
Realizam un nou tabel pentru depozit deoarece locatia depinde doar de acesta.

**FN2:**



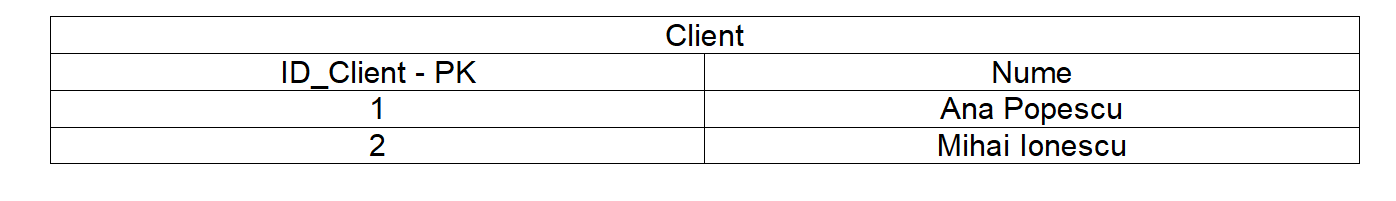
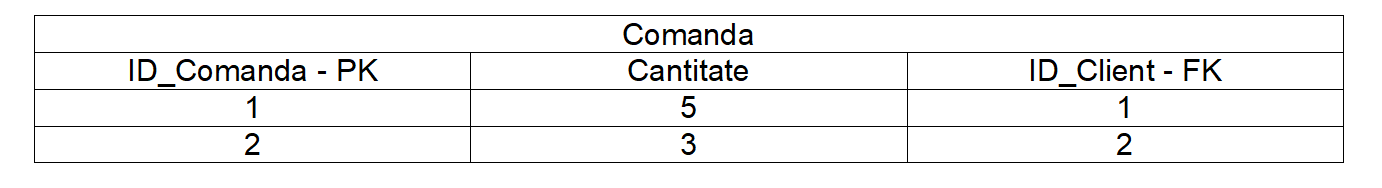


**NON FN3:**



Realizam un nou tabel pentru client deoarece nu depinde de cheia primara.

**FN3:**



10.Crearea unei secvente ce va fi utilizata in inserarea inregistrarilor in tabele.

CREATE SEQUENCE seq\_id\_material

START WITH 1

INCREMENT BY 1

NOCYCLE;



11.Crearea tabelelor în SQL și inserarea de date coerente în fiecare dintre acestea (minimum 5 înregistrări în fiecare tabel neasociativ; minimum 10 înregistrări în tabelele asociative).

--CREARE--

CREATE TABLE Client

(

ID\_Client INT NOT NULL,

NUME\_CLIENT VARCHAR(30) NOT NULL,

EMAIL\_CLIENT VARCHAR(10),

PRIMARY KEY (ID\_Client)

);

CREATE TABLE Locatie

(

ID\_Locatie VARCHAR(10) NOT NULL,

Tara VARCHAR(20) NOT NULL,

Oras VARCHAR(20) NOT NULL,

Strada VARCHAR(30),

Cod\_postal VARCHAR(10) NOT NULL

PRIMARY KEY (ID\_Locatie)

);

CREATE TABLE Magazin

(

ID\_Magazin VARCHAR(10) NOT NULL,

Nume VARCHAR(20) NOT NULL,

Telefon VARCHAR(20),

ID\_Locatie VARCHAR(10) NOT NULL,

PRIMARY KEY (ID\_Magazin),

FOREIGN KEY (ID\_Locatie) REFERENCES Locatie(ID\_Locatie)

);

create TABLE Angajat(

ID\_Angajat int not null primary key,

Nume\_Angajat varchar2(50) not null,

Salariu\_Angajat int not null,

Data\_Angajat date not null,

Telefon\_Angajat varchar2(10)

);

CREATE TABLE Angajat\_Magazin

(

ID\_Angajat\_Magazin INT NOT NULL,

Specializare VARCHAR(20) NOT NULL,

ID\_ANGAJAT not null,

ID\_Magazin VARCHAR(10) NOT NULL,

PRIMARY KEY (ID\_Angajat\_Magazin),

FOREIGN KEY (ID\_Angajat) REFERENCES Angajat(ID\_Angajat),

FOREIGN KEY (ID\_Magazin) REFERENCES Magazin(ID\_Magazin)

);

CREATE TABLE Fabrica

(

ID\_Fabrica VARCHAR(10) NOT NULL,

Productie INT NOT NULL,

Data\_infintare DATE NOT NULL,

ID\_Locatie VARCHAR(10) NOT NULL,

PRIMARY KEY (ID\_Fabrica),

FOREIGN KEY (ID\_Locatie) REFERENCES Locatie(ID\_Locatie)

);

CREATE TABLE Muncitor

(

ID\_Muncitor INT NOT NULL,

Specializare VARCHAR(20) NOT NULL,

ID\_Fabrica VARCHAR(10) NOT NULL,

ID\_Angajat not null,

PRIMARY KEY (ID\_Muncitor),

FOREIGN KEY (ID\_Angajat) REFERENCES Angajat(ID\_Angajat),

FOREIGN KEY (ID\_Fabrica) REFERENCES Fabrica(ID\_Fabrica)

);

CREATE TABLE Depozit

(

ID\_Depozit VARCHAR(10) NOT NULL,

Capacitate INT NOT NULL,

ID\_Locatie VARCHAR(10) NOT NULL,

PRIMARY KEY (ID\_Depozit),

FOREIGN KEY (ID\_Locatie) REFERENCES Locatie(ID\_Locatie)

);

CREATE TABLE Contract

(

ID\_Contract INT NOT NULL,

Suma INT NOT NULL,

ID\_Depozit VARCHAR(10) NOT NULL,

ID\_Magazin VARCHAR(10) NOT NULL,

PRIMARY KEY (ID\_Contract),

FOREIGN KEY (ID\_Depozit) REFERENCES Depozit(ID\_Depozit),

FOREIGN KEY (ID\_Magazin) REFERENCES Magazin(ID\_Magazin)

);

CREATE TABLE Geaca

(

ID\_Geaca VARCHAR(10) NOT NULL,

Nume VARCHAR(20) NOT NULL,

Tip VARCHAR(20) NOT NULL,

Marime VARCHAR(10) NOT NULL,

ID\_Fabrica VARCHAR(10) NOT NULL,

ID\_Depozit VARCHAR(10) NOT NULL,

PRIMARY KEY (ID\_Geaca),

FOREIGN KEY (ID\_Fabrica) REFERENCES Fabrica(ID\_Fabrica),

FOREIGN KEY (ID\_Depozit) REFERENCES Depozit(ID\_Depozit)

);

create TABLE Material(

ID\_Material int not null primary key,

Nume varchar2(50) not null,

Cantitate int

);

CREATE TABLE Construire (

ID\_Geaca VARCHAR(10) NOT NULL,

ID\_Material INT NOT NULL,

PRIMARY KEY (ID\_Geaca, ID\_Material),

FOREIGN KEY (ID\_Geaca) REFERENCES Geaca(ID\_Geaca),

FOREIGN KEY (ID\_Material) REFERENCES Materiale(ID\_Material)

);

CREATE TABLE Comanda

(

ID\_Comanda INT NOT NULL,

Cantitate INT NOT NULL,

Pret INT NOT NULL,

Data\_Comanda date,

ID\_Geaca VARCHAR(10) NOT NULL,

ID\_Magazin VARCHAR(10) NOT NULL,

ID\_Client INT NOT NULL,

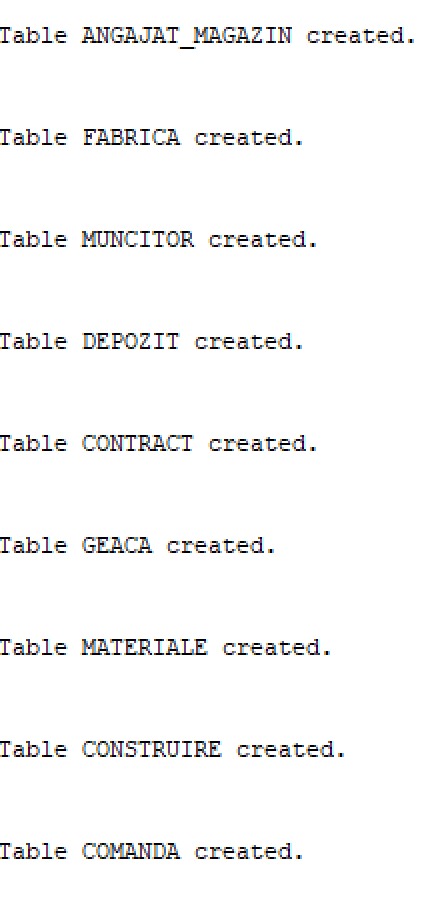
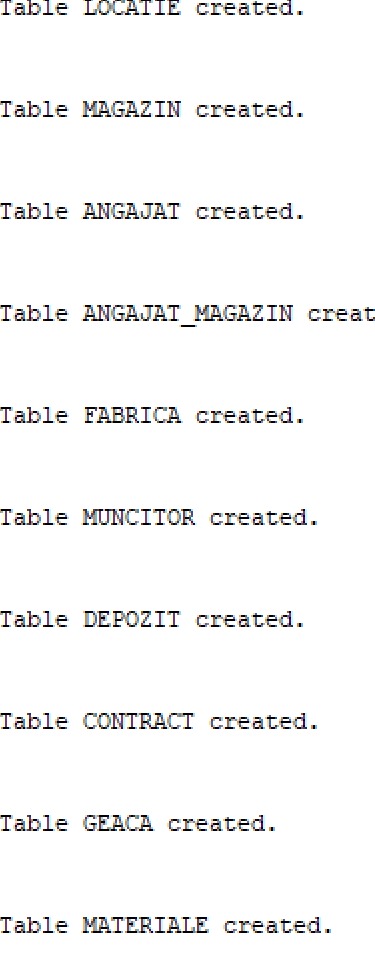
PRIMARY KEY (ID\_Comanda),

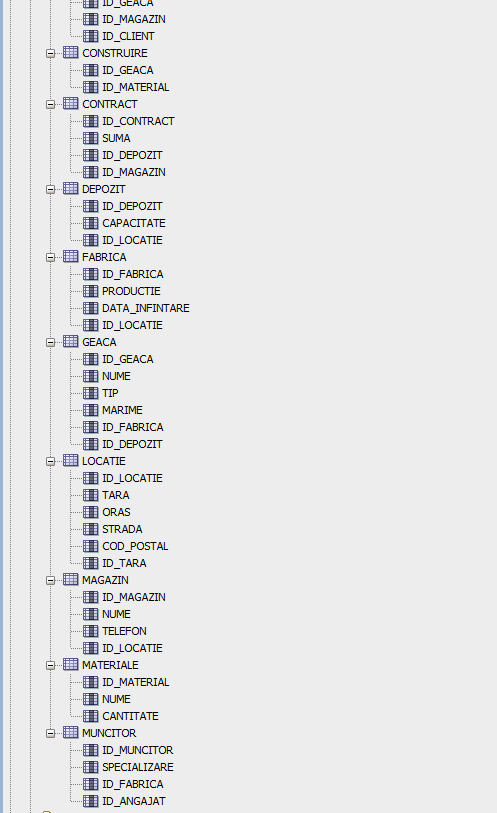
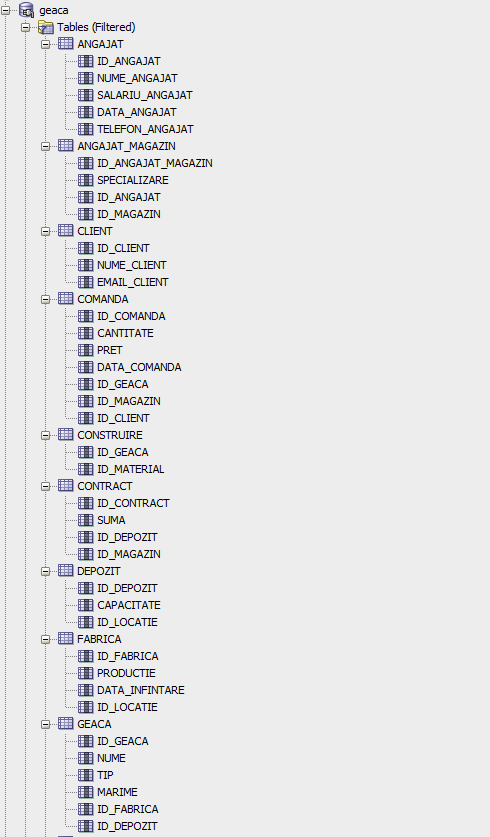
FOREIGN KEY (ID\_Geaca) REFERENCES Geaca(ID\_Geaca),

FOREIGN KEY (ID\_Magazin) REFERENCES Magazin(ID\_Magazin),

FOREIGN KEY (ID\_Client) REFERENCES Client(ID\_Client)

);





--INSERARE--

INSERT INTO Client (ID\_Client, NUME\_CLIENT, EMAIL\_CLIENT) VALUES (1, 'Ana Popescu', 'ana.popescu@example.com');

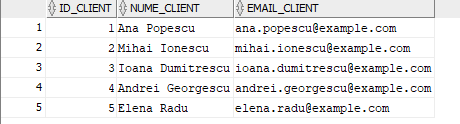
INSERT INTO Client (ID\_Client, NUME\_CLIENT, EMAIL\_CLIENT) VALUES(2, 'Mihai Ionescu', 'mihai.ionescu@example.com');

INSERT INTO Client (ID\_Client, NUME\_CLIENT, EMAIL\_CLIENT) VALUES(3, 'Ioana Dumitrescu', 'ioana.dumitrescu@example.com');

INSERT INTO Client (ID\_Client, NUME\_CLIENT, EMAIL\_CLIENT) VALUES(4, 'Andrei Georgescu', 'andrei.georgescu@example.com');

INSERT INTO Client (ID\_Client, NUME\_CLIENT, EMAIL\_CLIENT) VALUES(5, 'Elena Radu', 'elena.radu@example.com');

select \* from client;



INSERT INTO Locatie (ID\_Locatie, Tara, Oras, Strada, Cod\_postal) VALUES('RO001', 'Romania', 'Bucharest', 'Main Street', '010101');

INSERT INTO Locatie (ID\_Locatie, Tara, Oras, Strada, Cod\_postal) VALUES('RO002', 'Romania', 'Cluj-Napoca', 'Second Street', '020202');

INSERT INTO Locatie (ID\_Locatie, Tara, Oras, Strada, Cod\_postal) VALUES('RO003', 'Romania', 'Timisoara', 'Third Street', '030303');

INSERT INTO Locatie (ID\_Locatie, Tara, Oras, Strada, Cod\_postal) VALUES('RO004', 'Romania', 'Iasi', 'Fourth Street', '040404');

INSERT INTO Locatie (ID\_Locatie, Tara, Oras, Strada, Cod\_postal) VALUES('RO005', 'Romania', 'Constanta', 'Fifth Street', '050505');

INSERT INTO Locatie (ID\_Locatie, Tara, Oras, Strada, Cod\_postal) VALUES('RO006', 'Romania', 'Brasov', 'Sixth Street', '060606');

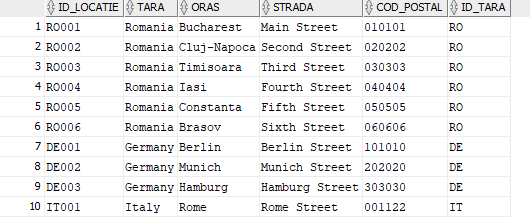
INSERT INTO Locatie (ID\_Locatie, Tara, Oras, Strada, Cod\_postal) VALUES ('DE001', 'Germany', 'Berlin', 'Berlin Street', '101010');

INSERT INTO Locatie (ID\_Locatie, Tara, Oras, Strada, Cod\_postal) VALUES('DE002', 'Germany', 'Munich', 'Munich Street', '202020');

INSERT INTO Locatie (ID\_Locatie, Tara, Oras, Strada, Cod\_postal) VALUES('DE003', 'Germany', 'Hamburg', 'Hamburg Street', '303030');

INSERT INTO Locatie (ID\_Locatie, Tara, Oras, Strada, Cod\_postal) VALUES('IT001', 'Italy', 'Rome', 'Rome Street', '001122', 'IT');

select \* from locatie;



INSERT INTO Magazin (ID\_Magazin, Nume, Telefon, ID\_Locatie) VALUES ('MAG001', 'Magazin1', '1234567890', 'RO001');

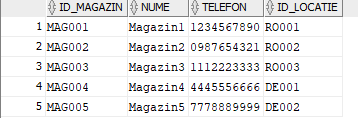
INSERT INTO Magazin (ID\_Magazin, Nume, Telefon, ID\_Locatie) VALUES('MAG002', 'Magazin2', '0987654321', 'RO002');

INSERT INTO Magazin (ID\_Magazin, Nume, Telefon, ID\_Locatie) VALUES('MAG003', 'Magazin3', '1112223333', 'RO003');

INSERT INTO Magazin (ID\_Magazin, Nume, Telefon, ID\_Locatie) VALUES('MAG004', 'Magazin4', '4445556666', 'DE001');

INSERT INTO Magazin (ID\_Magazin, Nume, Telefon, ID\_Locatie) VALUES('MAG005', 'Magazin5', '7778889999', 'DE002');

select \* from Magazin;



INSERT INTO Angajat (ID\_Angajat, Nume\_Angajat, Salariu\_Angajat, Data\_Angajat, Telefon\_Angajat) VALUES(1, 'Ion Popescu', 3000, TO\_DATE('2023-01-15', 'YYYY-MM-DD'), '0712345678');

INSERT INTO Angajat (ID\_Angajat, Nume\_Angajat, Salariu\_Angajat, Data\_Angajat, Telefon\_Angajat) VALUES(2, 'Maria Ionescu', 3500, TO\_DATE('2022-11-20', 'YYYY-MM-DD'), '0723456789');

INSERT INTO Angajat (ID\_Angajat, Nume\_Angajat, Salariu\_Angajat, Data\_Angajat, Telefon\_Angajat) VALUES(3, 'Ana Dumitrescu', 3200, TO\_DATE('2023-02-10', 'YYYY-MM-DD'), '0734567890');

INSERT INTO Angajat (ID\_Angajat, Nume\_Angajat, Salariu\_Angajat, Data\_Angajat, Telefon\_Angajat) VALUES(4, 'Andrei Georgescu', 4000, TO\_DATE('2023-03-05', 'YYYY-MM-DD'), '0745678901');

INSERT INTO Angajat (ID\_Angajat, Nume\_Angajat, Salariu\_Angajat, Data\_Angajat, Telefon\_Angajat) VALUES(5, 'Elena Radu', 3800, TO\_DATE('2022-09-25', 'YYYY-MM-DD'), '0756789012');

INSERT INTO Angajat (ID\_Angajat, Nume\_Angajat, Salariu\_Angajat, Data\_Angajat, Telefon\_Angajat) VALUES(6, 'Mihai Stan', 3400, TO\_DATE('2022-12-10', 'YYYY-MM-DD'), '0767890123');

INSERT INTO Angajat (ID\_Angajat, Nume\_Angajat, Salariu\_Angajat, Data\_Angajat, Telefon\_Angajat) VALUES(7, 'Cristina Stoica', 3700, TO\_DATE('2023-04-15', 'YYYY-MM-DD'), '0778901234');

INSERT INTO Angajat (ID\_Angajat, Nume\_Angajat, Salariu\_Angajat, Data\_Angajat, Telefon\_Angajat) VALUES(8, 'Alexandra Marin', 3600, TO\_DATE('2023-05-20', 'YYYY-MM-DD'), '0789012345');

INSERT INTO Angajat (ID\_Angajat, Nume\_Angajat, Salariu\_Angajat, Data\_Angajat, Telefon\_Angajat) VALUES(9, 'Florin Munteanu', 3900, TO\_DATE('2023-07-01', 'YYYY-MM-DD'), '0790123456');

INSERT INTO Angajat (ID\_Angajat, Nume\_Angajat, Salariu\_Angajat, Data\_Angajat, Telefon\_Angajat) VALUES(10, 'Diana Andreescu', 3300, TO\_DATE('2023-08-10', 'YYYY-MM-DD'), '0801234567');

INSERT INTO Angajat (ID\_Angajat, Nume\_Angajat, Salariu\_Angajat, Data\_Angajat, Telefon\_Angajat) VALUES(11, 'Gabriel Popa', 3100, TO\_DATE('2023-09-15', 'YYYY-MM-DD'), '0812345678');

INSERT INTO Angajat (ID\_Angajat, Nume\_Angajat, Salariu\_Angajat, Data\_Angajat, Telefon\_Angajat) VALUES(12, 'Andreea Dumitrascu', 3400, TO\_DATE('2023-10-20', 'YYYY-MM-DD'), '0823456789');

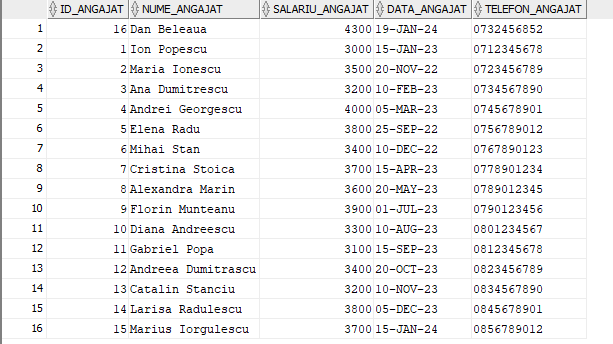
INSERT INTO Angajat (ID\_Angajat, Nume\_Angajat, Salariu\_Angajat, Data\_Angajat, Telefon\_Angajat) VALUES(13, 'Catalin Stanciu', 3200, TO\_DATE('2023-11-10', 'YYYY-MM-DD'), '0834567890');

INSERT INTO Angajat (ID\_Angajat, Nume\_Angajat, Salariu\_Angajat, Data\_Angajat, Telefon\_Angajat) VALUES(14, 'Larisa Radulescu', 3800, TO\_DATE('2023-12-05', 'YYYY-MM-DD'), '0845678901');

INSERT INTO Angajat (ID\_Angajat, Nume\_Angajat, Salariu\_Angajat, Data\_Angajat, Telefon\_Angajat) VALUES(15, 'Marius Iorgulescu', 3700, TO\_DATE('2024-01-15', 'YYYY-MM-DD'), '0856789012');

INSERT INTO Angajat (ID\_Angajat, Nume\_Angajat, Salariu\_Angajat, Data\_Angajat, Telefon\_Angajat) VALUES(16, 'Dan Beleaua', 4300, TO\_DATE('2024-01-19', 'YYYY-MM-DD'), '0732456852');

select \* from angajat;



INSERT INTO Angajat\_Magazin (ID\_Angajat\_Magazin, Specializare, ID\_Angajat, ID\_Magazin) VALUES(1, 'Vanzator', 1, 'MAG001');

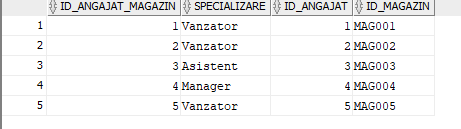
INSERT INTO Angajat\_Magazin (ID\_Angajat\_Magazin, Specializare, ID\_Angajat, ID\_Magazin) VALUES(2, 'Vanzator', 2, 'MAG002');

INSERT INTO Angajat\_Magazin (ID\_Angajat\_Magazin, Specializare, ID\_Angajat, ID\_Magazin) VALUES(3, 'Asistent', 3, 'MAG003');

INSERT INTO Angajat\_Magazin (ID\_Angajat\_Magazin, Specializare, ID\_Angajat, ID\_Magazin) VALUES(4, 'Manager', 4, 'MAG004');

INSERT INTO Angajat\_Magazin (ID\_Angajat\_Magazin, Specializare, ID\_Angajat, ID\_Magazin) VALUES(5, 'Vanzator', 5, 'MAG005');

select \* from angajat\_magazin;



INSERT INTO Fabrica (ID\_Fabrica, Productie, Data\_infintare, ID\_Locatie) VALUES('FAB001', 1000, TO\_DATE('2020-01-01', 'YYYY-MM-DD'), 'RO001');

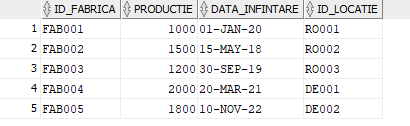
INSERT INTO Fabrica (ID\_Fabrica, Productie, Data\_infintare, ID\_Locatie) VALUES('FAB002', 1500, TO\_DATE('2018-05-15', 'YYYY-MM-DD'), 'RO002');

INSERT INTO Fabrica (ID\_Fabrica, Productie, Data\_infintare, ID\_Locatie) VALUES('FAB003', 1200, TO\_DATE('2019-09-30', 'YYYY-MM-DD'), 'RO003');

INSERT INTO Fabrica (ID\_Fabrica, Productie, Data\_infintare, ID\_Locatie) VALUES('FAB004', 2000, TO\_DATE('2021-03-20', 'YYYY-MM-DD'), 'DE001');

INSERT INTO Fabrica (ID\_Fabrica, Productie, Data\_infintare, ID\_Locatie) VALUES('FAB005', 1800, TO\_DATE('2022-11-10', 'YYYY-MM-DD'), 'DE002');

select \* from fabrica;



INSERT INTO Muncitor (ID\_Muncitor, Specializare, ID\_Fabrica, ID\_Angajat) VALUES(6, 'Croitor', 'FAB001', 6);

INSERT INTO Muncitor (ID\_Muncitor, Specializare, ID\_Fabrica, ID\_Angajat) VALUES(7, 'Croitor', 'FAB002', 7);

INSERT INTO Muncitor (ID\_Muncitor, Specializare, ID\_Fabrica, ID\_Angajat) VALUES(8, 'Croitor', 'FAB003', 8);

INSERT INTO Muncitor (ID\_Muncitor, Specializare, ID\_Fabrica, ID\_Angajat) VALUES(9, 'Croitor', 'FAB004', 9);

INSERT INTO Muncitor (ID\_Muncitor, Specializare, ID\_Fabrica, ID\_Angajat) VALUES(10, 'Croitor', 'FAB005', 10);

INSERT INTO Muncitor (ID\_Muncitor, Specializare, ID\_Fabrica, ID\_Angajat) VALUES(11, 'Cusator', 'FAB001', 11);

INSERT INTO Muncitor (ID\_Muncitor, Specializare, ID\_Fabrica, ID\_Angajat) VALUES(12, 'Cusator', 'FAB002', 12);

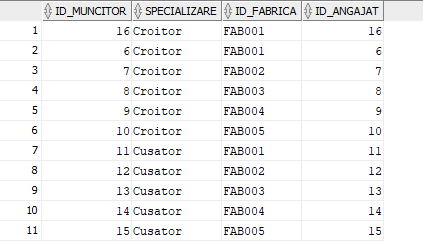
INSERT INTO Muncitor (ID\_Muncitor, Specializare, ID\_Fabrica, ID\_Angajat) VALUES(13, 'Cusator', 'FAB003', 13);

INSERT INTO Muncitor (ID\_Muncitor, Specializare, ID\_Fabrica, ID\_Angajat) VALUES(14, 'Cusator', 'FAB004', 14);

INSERT INTO Muncitor (ID\_Muncitor, Specializare, ID\_Fabrica, ID\_Angajat) VALUES(15, 'Cusator', 'FAB005', 15);

INSERT INTO Muncitor (ID\_Muncitor, Specializare, ID\_Fabrica, ID\_Angajat) VALUES(16, 'Croitor', 'FAB001', 16);

select \* from muncitor;



INSERT INTO Depozit (ID\_Depozit, Capacitate, ID\_Locatie) VALUES('DEP001', 1000, 'RO001');

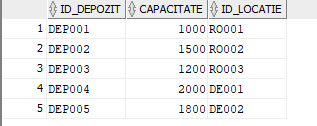
INSERT INTO Depozit (ID\_Depozit, Capacitate, ID\_Locatie) VALUES('DEP002', 1500, 'RO002');

INSERT INTO Depozit (ID\_Depozit, Capacitate, ID\_Locatie) VALUES('DEP003', 1200, 'RO003');

INSERT INTO Depozit (ID\_Depozit, Capacitate, ID\_Locatie) VALUES('DEP004', 2000, 'DE001');

INSERT INTO Depozit (ID\_Depozit, Capacitate, ID\_Locatie) VALUES('DEP005', 1800, 'DE002');

select \* from depozit;



INSERT INTO Contract (ID\_Contract, Suma, ID\_Depozit, ID\_Magazin) VALUES(1, 5000, 'DEP001', 'MAG001');

INSERT INTO Contract (ID\_Contract, Suma, ID\_Depozit, ID\_Magazin) VALUES(2, 6000, 'DEP002', 'MAG002');

INSERT INTO Contract (ID\_Contract, Suma, ID\_Depozit, ID\_Magazin) VALUES(3, 7000, 'DEP003', 'MAG003');

INSERT INTO Contract (ID\_Contract, Suma, ID\_Depozit, ID\_Magazin) VALUES(4, 8000, 'DEP004', 'MAG004');

INSERT INTO Contract (ID\_Contract, Suma, ID\_Depozit, ID\_Magazin) VALUES(5, 9000, 'DEP005', 'MAG005');

INSERT INTO Contract (ID\_Contract, Suma, ID\_Depozit, ID\_Magazin) VALUES(6, 5500, 'DEP001', 'MAG001');

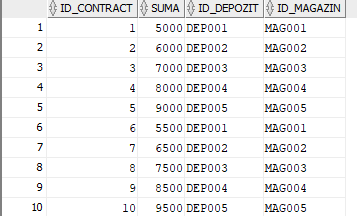
INSERT INTO Contract (ID\_Contract, Suma, ID\_Depozit, ID\_Magazin) VALUES(7, 6500, 'DEP002', 'MAG002');

INSERT INTO Contract (ID\_Contract, Suma, ID\_Depozit, ID\_Magazin) VALUES(8, 7500, 'DEP003', 'MAG003');

INSERT INTO Contract (ID\_Contract, Suma, ID\_Depozit, ID\_Magazin) VALUES(9, 8500, 'DEP004', 'MAG004');

INSERT INTO Contract (ID\_Contract, Suma, ID\_Depozit, ID\_Magazin) VALUES(10, 9500, 'DEP005', 'MAG005');

select \* from contract;



INSERT INTO Geaca (ID\_Geaca, Nume, Tip, Marime, ID\_Fabrica, ID\_Depozit) VALUES('G001', 'Winter Jacket', 'Winter', 'M', 'FAB001', 'DEP001');

INSERT INTO Geaca (ID\_Geaca, Nume, Tip, Marime, ID\_Fabrica, ID\_Depozit) VALUES('G002', 'Leather Jacket', 'Leather', 'L', 'FAB002', 'DEP002');

INSERT INTO Geaca (ID\_Geaca, Nume, Tip, Marime, ID\_Fabrica, ID\_Depozit) VALUES('G003', 'Sport Jacket', 'Sport', 'S', 'FAB003', 'DEP003');

INSERT INTO Geaca (ID\_Geaca, Nume, Tip, Marime, ID\_Fabrica, ID\_Depozit) VALUES('G004', 'Denim Jacket', 'Denim', 'XL', 'FAB004', 'DEP004');

INSERT INTO Geaca (ID\_Geaca, Nume, Tip, Marime, ID\_Fabrica, ID\_Depozit) VALUES('G005', 'Parka Jacket', 'Parka', 'M', 'FAB005', 'DEP005');

INSERT INTO Geaca (ID\_Geaca, Nume, Tip, Marime, ID\_Fabrica, ID\_Depozit) VALUES('G006', 'Bomber Jacket', 'Bomber', 'L', 'FAB001', 'DEP001');

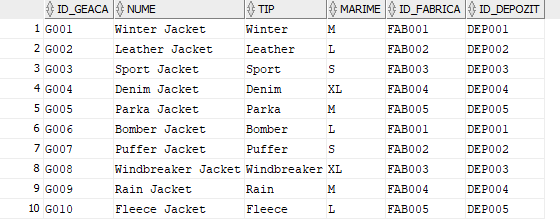
INSERT INTO Geaca (ID\_Geaca, Nume, Tip, Marime, ID\_Fabrica, ID\_Depozit) VALUES('G007', 'Puffer Jacket', 'Puffer', 'S', 'FAB002', 'DEP002');

INSERT INTO Geaca (ID\_Geaca, Nume, Tip, Marime, ID\_Fabrica, ID\_Depozit) VALUES('G008', 'Windbreaker Jacket', 'Windbreaker', 'XL', 'FAB003', 'DEP003');

INSERT INTO Geaca (ID\_Geaca, Nume, Tip, Marime, ID\_Fabrica, ID\_Depozit) VALUES('G009', 'Rain Jacket', 'Rain', 'M', 'FAB004', 'DEP004');

INSERT INTO Geaca (ID\_Geaca, Nume, Tip, Marime, ID\_Fabrica, ID\_Depozit) VALUES('G010', 'Fleece Jacket', 'Fleece', 'L', 'FAB005', 'DEP005');

select \* from geaca;



INSERT INTO Comanda (ID\_Comanda, Cantitate, Pret, Data\_Comanda, ID\_Geaca, ID\_Magazin, ID\_Client) VALUES(1, 5, 200, TO\_DATE('2024-03-01', 'YYYY-MM-DD'), 'G001', 'MAG001', 1);

INSERT INTO Comanda (ID\_Comanda, Cantitate, Pret, Data\_Comanda, ID\_Geaca, ID\_Magazin, ID\_Client) VALUES(2, 3, 300, TO\_DATE('2023-03-01', 'YYYY-MM-DD'), 'G002', 'MAG002', 2);

INSERT INTO Comanda (ID\_Comanda, Cantitate, Pret, Data\_Comanda, ID\_Geaca, ID\_Magazin, ID\_Client) VALUES(3, 7, 250, TO\_DATE('2024-01-05', 'YYYY-MM-DD'), 'G003', 'MAG003', 3);

INSERT INTO Comanda (ID\_Comanda, Cantitate, Pret, Data\_Comanda, ID\_Geaca, ID\_Magazin, ID\_Client) VALUES(4, 4, 180, TO\_DATE('2024-01-19', 'YYYY-MM-DD'), 'G004', 'MAG004', 4);

INSERT INTO Comanda (ID\_Comanda, Cantitate, Pret, Data\_Comanda, ID\_Geaca, ID\_Magazin, ID\_Client) VALUES(5, 6, 220, TO\_DATE('2023-08-29', 'YYYY-MM-DD'), 'G005', 'MAG005', 5);

INSERT INTO Comanda (ID\_Comanda, Cantitate, Pret, Data\_Comanda, ID\_Geaca, ID\_Magazin, ID\_Client) VALUES(6, 2, 150, TO\_DATE('2022-04-10', 'YYYY-MM-DD'), 'G006', 'MAG001', 3);

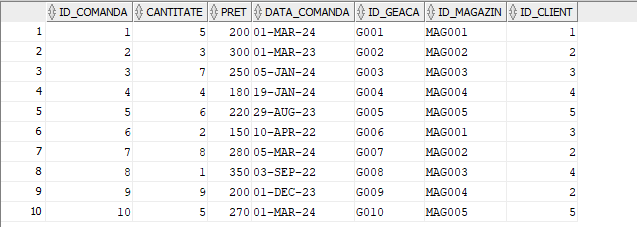
INSERT INTO Comanda (ID\_Comanda, Cantitate, Pret, Data\_Comanda, ID\_Geaca, ID\_Magazin, ID\_Client) VALUES(7, 8, 280, TO\_DATE('2024-03-5', 'YYYY-MM-DD'), 'G007', 'MAG002', 2);

INSERT INTO Comanda (ID\_Comanda, Cantitate, Pret, Data\_Comanda, ID\_Geaca, ID\_Magazin, ID\_Client) VALUES(8, 1, 350, TO\_DATE('2022-9-03', 'YYYY-MM-DD'), 'G008', 'MAG003', 4);

INSERT INTO Comanda (ID\_Comanda, Cantitate, Pret, Data\_Comanda, ID\_Geaca, ID\_Magazin, ID\_Client) VALUES(9, 9, 200, TO\_DATE('2023-12-01', 'YYYY-MM-DD'), 'G009', 'MAG004', 2);

INSERT INTO Comanda (ID\_Comanda, Cantitate, Pret, Data\_Comanda, ID\_Geaca, ID\_Magazin, ID\_Client) VALUES(10, 5, 270, TO\_DATE('2024-03-01', 'YYYY-MM-DD'), 'G010', 'MAG005', 5);

select \* from comanda;



INSERT INTO Materiale (ID\_Material, Nume, Cantitate) VALUES(seq\_id\_material.NEXTVAL, 'Bumbac', 100);

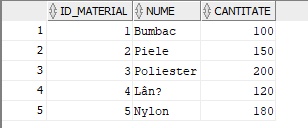
INSERT INTO Materiale (ID\_Material, Nume, Cantitate) VALUES(seq\_id\_material.NEXTVAL, 'Piele', 150);

INSERT INTO Materiale (ID\_Material, Nume, Cantitate) VALUES(seq\_id\_material.NEXTVAL, 'Poliester', 200);

INSERT INTO Materiale (ID\_Material, Nume, Cantitate) VALUES(seq\_id\_material.NEXTVAL, 'Lân?', 120);

INSERT INTO Materiale (ID\_Material, Nume, Cantitate) VALUES(seq\_id\_material.NEXTVAL, 'Nylon', 180);

select \* from materiale;



INSERT INTO Construire (ID\_Geaca, ID\_Material) VALUES('G001', 1);

INSERT INTO Construire (ID\_Geaca, ID\_Material) VALUES('G001', 2);

INSERT INTO Construire (ID\_Geaca, ID\_Material) VALUES('G002', 3);

INSERT INTO Construire (ID\_Geaca, ID\_Material) VALUES('G003', 4);

INSERT INTO Construire (ID\_Geaca, ID\_Material) VALUES('G004', 5);

INSERT INTO Construire (ID\_Geaca, ID\_Material) VALUES('G005', 3);

INSERT INTO Construire (ID\_Geaca, ID\_Material) VALUES('G005', 3);

INSERT INTO Construire (ID\_Geaca, ID\_Material) VALUES('G002', 1);

INSERT INTO Construire (ID\_Geaca, ID\_Material) VALUES('G005', 4);

INSERT INTO Construire (ID\_Geaca, ID\_Material) VALUES('G004', 2);

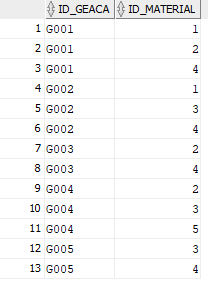
INSERT INTO Construire (ID\_Geaca, ID\_Material) VALUES('G004', 3);

INSERT INTO Construire (ID\_Geaca, ID\_Material) VALUES('G003', 2);

INSERT INTO Construire (ID\_Geaca, ID\_Material) VALUES('G001', 4);

INSERT INTO Construire (ID\_Geaca, ID\_Material) VALUES('G002', 4);

select \* from construire;



12.Formulați în limbaj natural și implementați 5 cereri SQL complexe ce vor utiliza, în ansamblul lor, următoarele elemente:

* + 1. • subcereri sincronizate în care intervin cel puțin 3 tabele
    2. • subcereri nesincronizate în clauza FROM
    3. • grupări de date cu subcereri nesincronizate in care intervin cel putin 3 tabele, funcții grup, filtrare la nivel de grupuri (in cadrul aceleiasi cereri) • ordonări si utilizarea funcțiilor NVL și DECODE (in cadrul aceleiasi cereri)
    4. • utilizarea a cel puțin 2 funcții pe șiruri de caractere, 2 funcții pe date calendaristice, a cel puțin unei expresii CASE
    5. • utilizarea a cel puțin 1 bloc de cerere (clauza WITH)

**12.1 Exercitiul 1**

**Cerere cu grupari de date, functii grup, filtrare la nivel de grupuri cu subcereri nesincronizate (in clauza having) in care intervin cel putin 3 tabele.**

Sa se afiseze numele clientilor si numarul de geci diferite pe care le-au comandat.Selecteaza doar clientii care au cumparat un numar de geci diferite mai mare decat media numarului de geci diferite cumparate de toti clientii.Rezultatele vor fi ordonate descrescator dupa nr de geci diferite.

SELECT

C.NUME\_CLIENT,

COUNT(DISTINCT G.ID\_Geaca) AS Numar\_Geci\_Diferite

FROM

Client C

JOIN

Comanda CM ON C.ID\_Client = CM.ID\_Client

JOIN

Geaca G ON CM.ID\_Geaca = G.ID\_Geaca

JOIN

Magazin M ON CM.ID\_Magazin = M.ID\_Magazin

GROUP BY

C.NUME\_CLIENT

HAVING

COUNT(DISTINCT G.ID\_Geaca) > (

SELECT AVG(Numar\_Geci)

FROM (

SELECT

CM2.ID\_Client,

COUNT(DISTINCT G2.ID\_Geaca) AS Numar\_Geci

FROM

Comanda CM2

JOIN

Geaca G2 ON CM2.ID\_Geaca = G2.ID\_Geaca

JOIN

Magazin M2 ON CM2.ID\_Magazin = M2.ID\_Magazin

GROUP BY

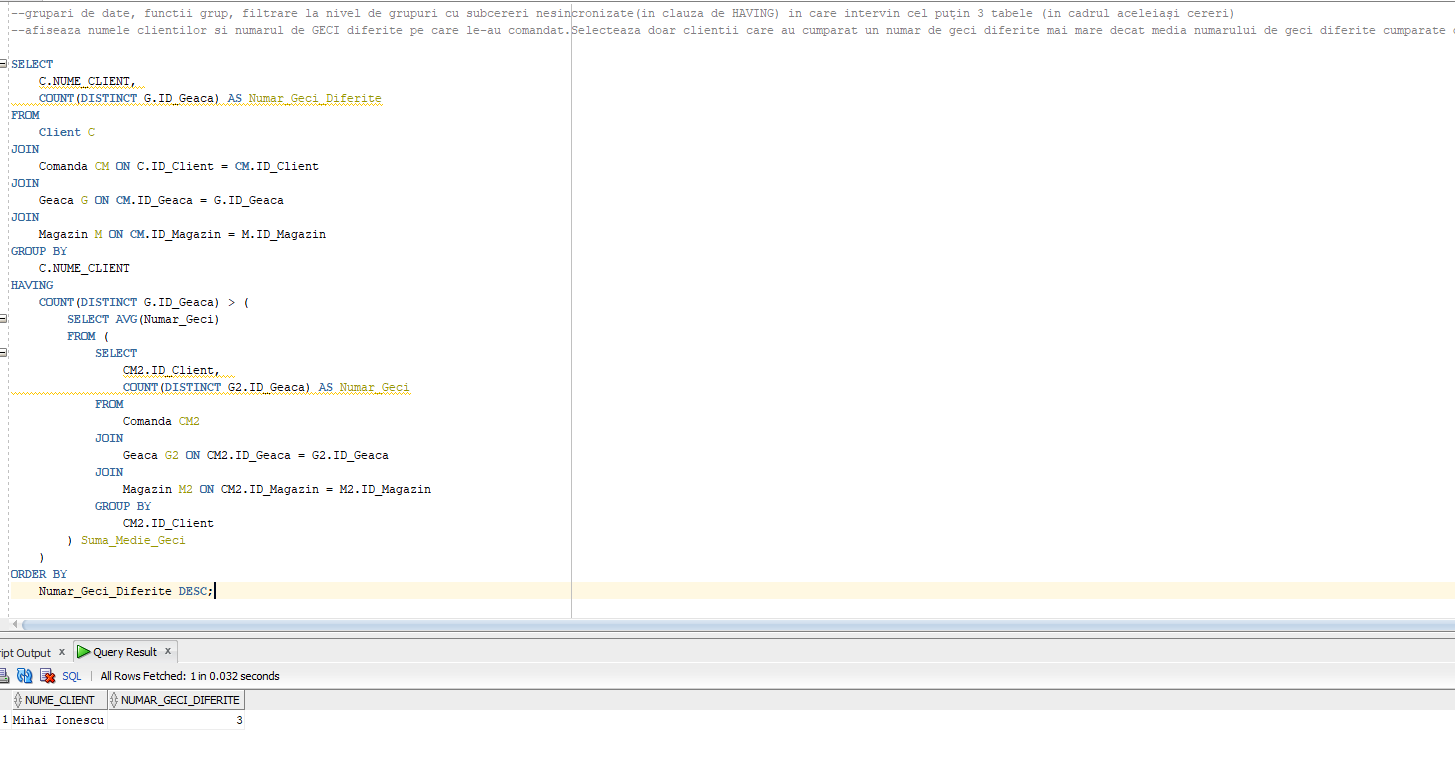
CM2.ID\_Client

) Suma\_Medie\_Geci

)

ORDER BY

Numar\_Geci\_Diferite DESC;



**12.2 Exercitiul 2**

**Subcereri nesincronizate in clauza from, expresia CASE.**

Sa se afiseze numele si email-ul pt fiecare client.Clientii care au comenzi mai mari de 500 de lei devin "Clienti premium", iar ceilalti "Clienti uzuali".

SELECT

C.Nume\_Client,

C.Email\_Client,

CASE

WHEN NVL(Suma\_Comenzi, 0) > 500 THEN 'Client premium'

ELSE 'Client uzual'

END AS Tip\_Client

FROM

Client C

LEFT JOIN

(

SELECT

ID\_Client,

SUM(Pret) AS Suma\_Comenzi

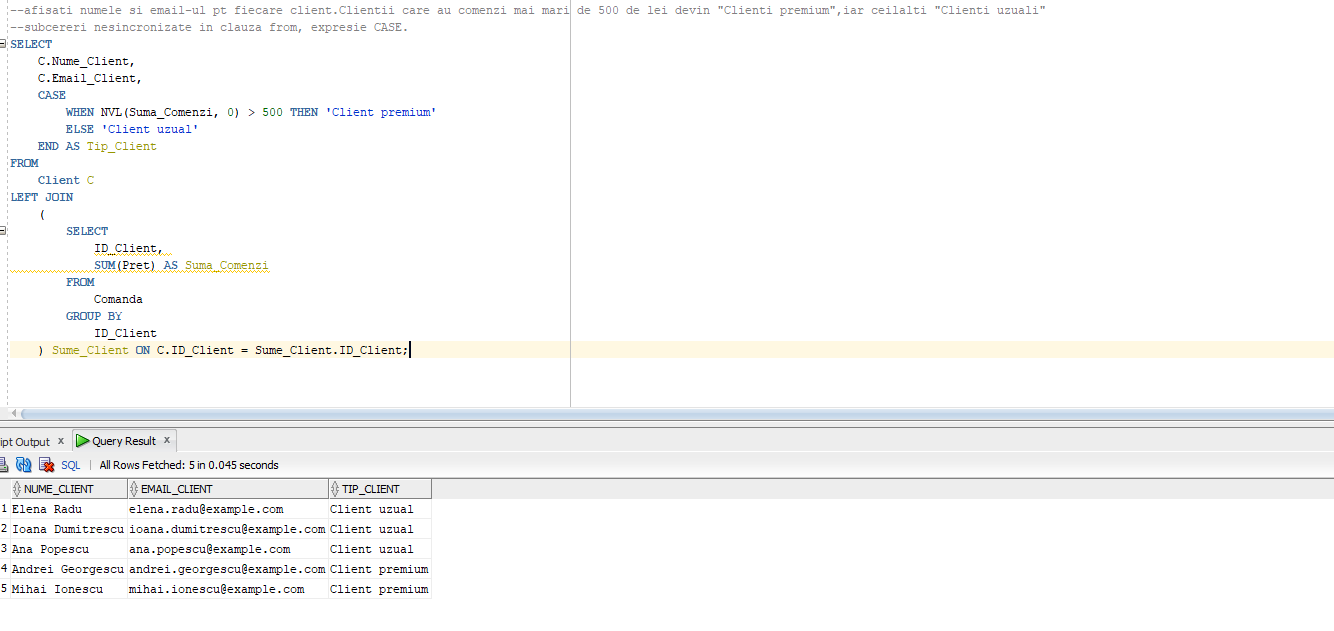
FROM

Comanda

GROUP BY

ID\_Client

) Sume\_Client ON C.ID\_Client = Sume\_Client.ID\_Client;



**12.3 Exercitiul 3**

**NVL, DECODE, ORDER BY**

Sa se afiseze numele , id-ul si numele magazinului in care lucreaza , si salariul vechi si majorat stiind ca se majoreaza astfel: 20% pt Magazin1,15% pentru Magazin2, 10% pentru Magazin3, 10 % pentru Magazin4 si 20% pentru Magazin5. Salariul este majorat angajatilor care s-au angajat dupa 2022 si se afiseaza in ordinea magazinelor in functie de id.

SELECT

A.Nume\_Angajat,

A.ID\_Angajat,

M.Nume,

A.Salariu\_Angajat AS Salariu\_Vechi,

NVL(

DECODE(

M.ID\_Magazin,

'MAG001', A.Salariu\_Angajat \* 1.20,

'MAG002', A.Salariu\_Angajat \* 1.15,

'MAG003', A.Salariu\_Angajat \* 1.10,

'MAG004', A.Salariu\_Angajat \* 1.10,

'MAG005', A.Salariu\_Angajat \* 1.20

),

A.Salariu\_Angajat

) AS Salariu\_Majorat

FROM

Angajat A

INNER JOIN

Angajat\_Magazin AM ON A.ID\_Angajat = AM.ID\_Angajat

INNER JOIN

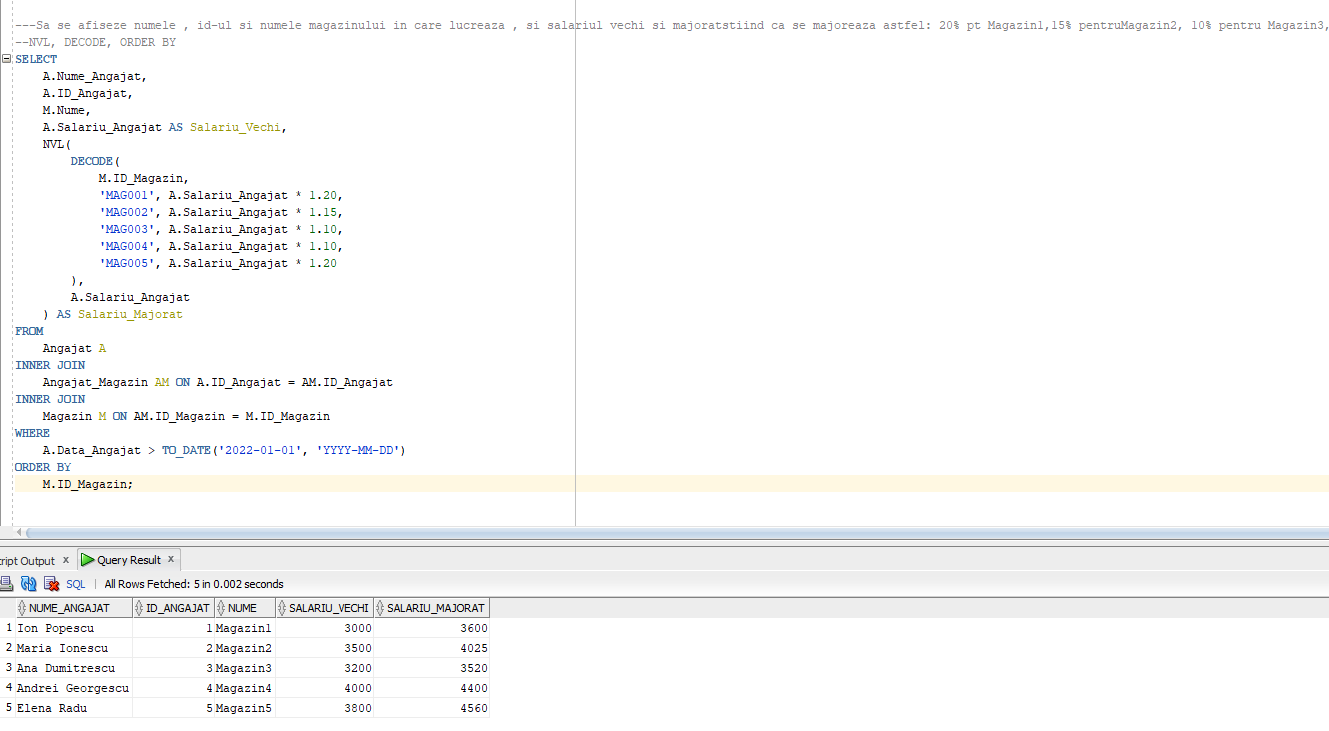
Magazin M ON AM.ID\_Magazin = M.ID\_Magazin

WHERE

A.Data\_Angajat > TO\_DATE('2022-01-01', 'YYYY-MM-DD')

ORDER BY

1. ID\_Magazin;



**12.4 Exercitiul 4**

**Subcereri sincronizate in care intervin cel putin 3 tabele, functii pe date calendaristice.**

Pentru toate comenzile plasate in anul 2023 sa se afiseze urmatoarele informatii: numele clientului, numele magazinului, data in care a fost efectuata comanda, numele gecii comandate, respectiv numarul de ani si luni trecuti de la efectuarea comenzii.

SELECT

c.NUME\_CLIENT,

m.Nume AS NUME\_MAGAZIN,

TO\_CHAR(o.Data\_Comanda, 'DD MONTH YY') AS DATA\_COMANDA,

g.Nume AS NUME\_GEACA,

f.ID\_Fabrica AS NUME\_FABRICA,

TRUNC(MONTHS\_BETWEEN(SYSDATE, o.Data\_Comanda) / 12) AS ANI\_TRECUTI,

MOD(TRUNC(MONTHS\_BETWEEN(SYSDATE, o.Data\_Comanda)), 12) AS LUNI\_TRECUTI

FROM

Comanda o

JOIN Client c ON o.ID\_Client = c.ID\_Client

JOIN Magazin m ON o.ID\_Magazin = m.ID\_Magazin

JOIN Geaca g ON o.ID\_Geaca = g.ID\_Geaca

JOIN Fabrica f ON g.ID\_Fabrica = f.ID\_Fabrica

WHERE

o.ID\_Client IN (

SELECT

o2.ID\_Client

FROM

Comanda o2

JOIN Geaca g2 ON o2.ID\_Geaca = g2.ID\_Geaca

JOIN Fabrica f2 ON g2.ID\_Fabrica = f2.ID\_Fabrica

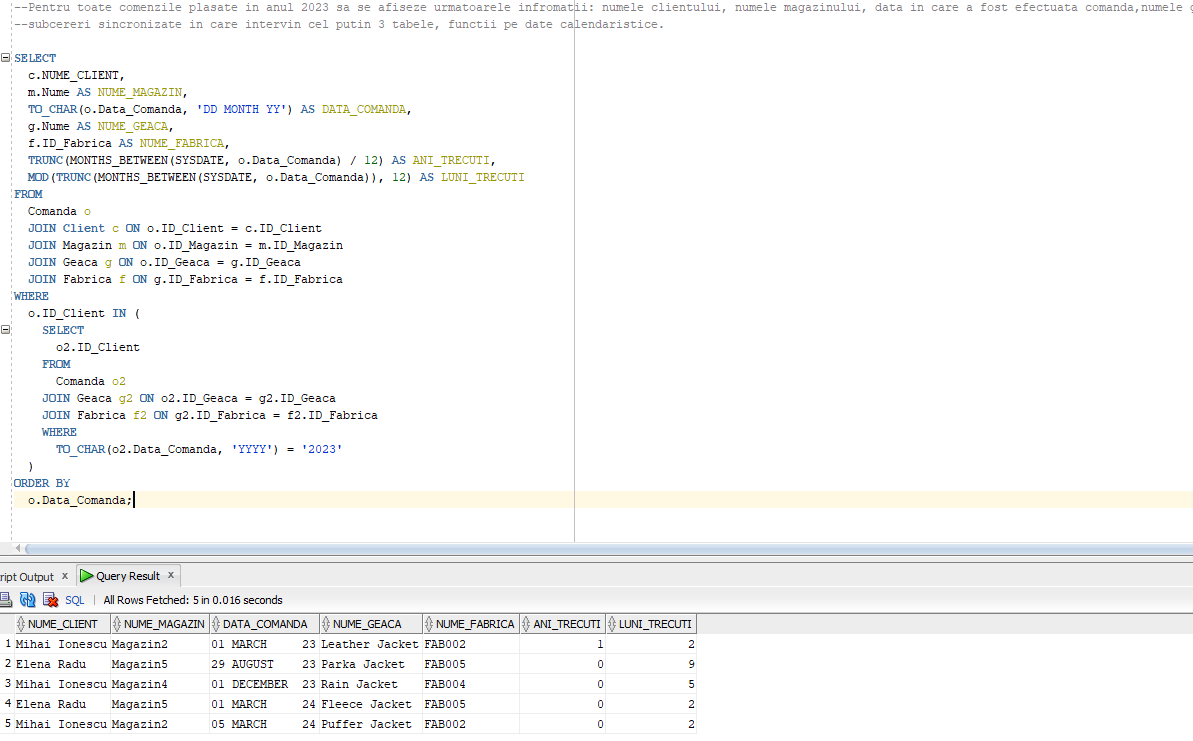
WHERE

TO\_CHAR(o2.Data\_Comanda, 'YYYY') = '2023'

)

ORDER BY

o.Data\_Comanda;



**12.5 Exercitiul 5**

**Clauza WITH, functii pe siruri de caractere**

Sa se afiseze numele angajatilor si numele magazinelor care au cel putin o comanda plasata de un client cu un email care se termina cu ".com", si care au produs cel putin o geaca de tip "Polar" in fabrica cu ID-ul "FAB001".

WITH Comenzi\_Email AS (

SELECT DISTINCT Co.ID\_Magazin

FROM Comanda Co

JOIN Client Cl ON Co.ID\_Client = Cl.ID\_Client

WHERE Cl.EMAIL\_CLIENT LIKE '%.com'

),

Geaci\_Polare AS (

SELECT DISTINCT G.ID\_Geaca

FROM Geaca G

WHERE LOWER(G.Tip) = 'winter'

AND UPPER(G.ID\_Fabrica) = 'FAB001'

)

SELECT DISTINCT A.Nume\_Angajat, M.Nume AS Nume\_Magazin

FROM Angajat A

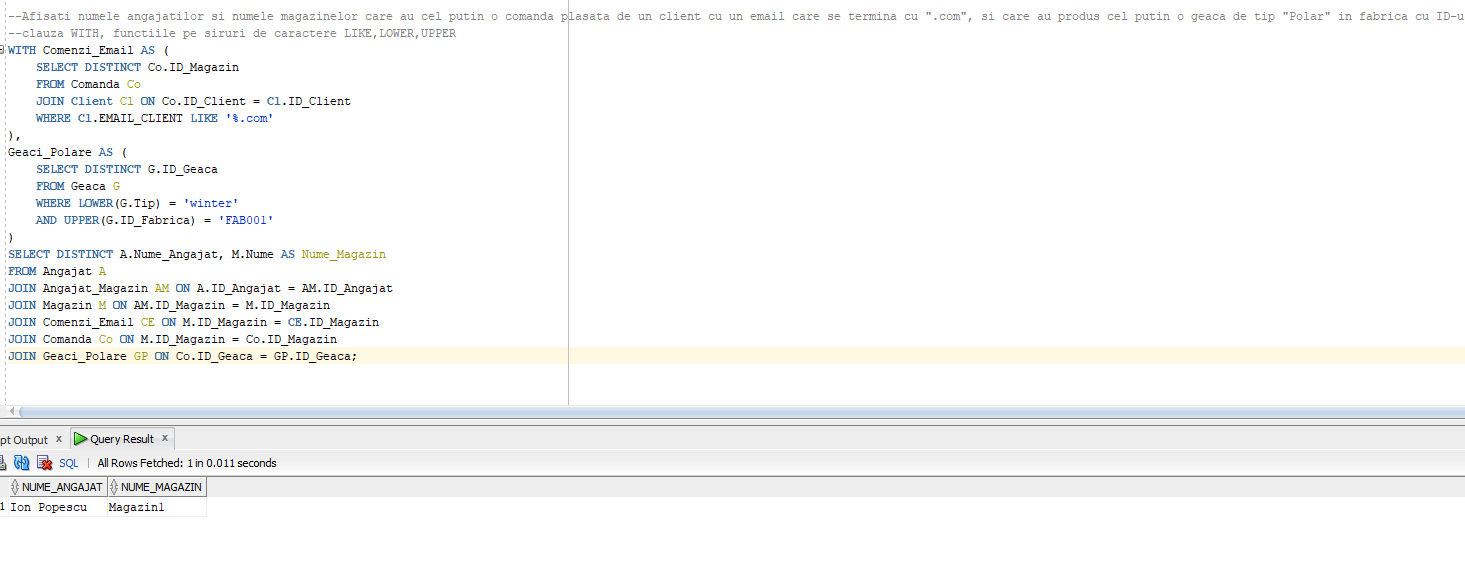
JOIN Angajat\_Magazin AM ON A.ID\_Angajat = AM.ID\_Angajat

JOIN Magazin M ON AM.ID\_Magazin = M.ID\_Magazin

JOIN Comenzi\_Email CE ON M.ID\_Magazin = CE.ID\_Magazin

JOIN Comanda Co ON M.ID\_Magazin = Co.ID\_Magazin

JOIN Geaci\_Polare GP ON Co.ID\_Geaca = GP.ID\_Geaca;



13.Implementarea a 3 operații de actualizare și de suprimare a datelor utilizând subcereri.

1.Aplicati o reducere de 20% la toate comenzile care sunt plasate de un client care a plasat 2 sau mai multe comenzi.

update Comanda

set Pret=Pret-Pret/10

where ID\_Client in

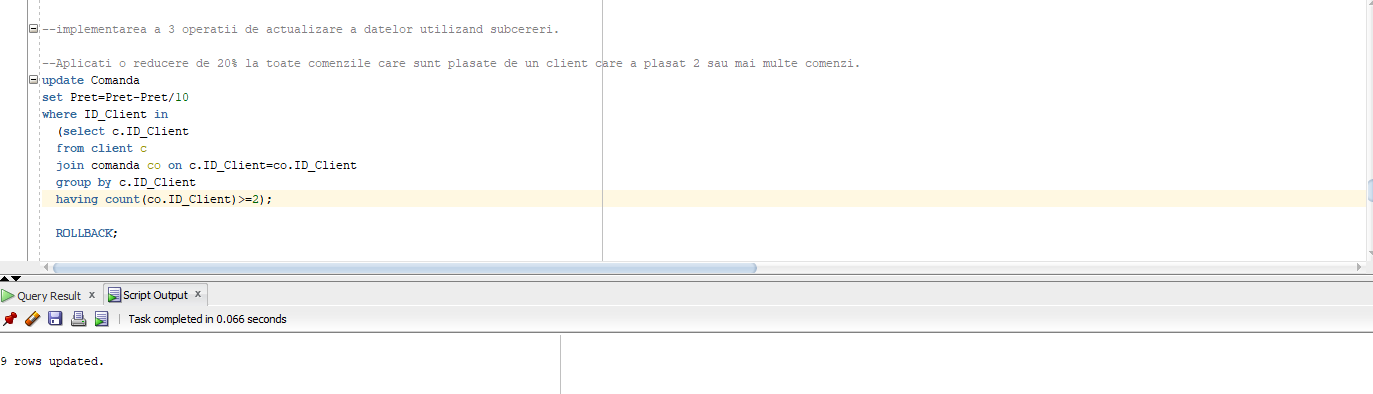
(select c.ID\_Client

from client c

join comanda co on c.ID\_Client=co.ID\_Client

group by c.ID\_Client

having count(co.ID\_Client)>=2);

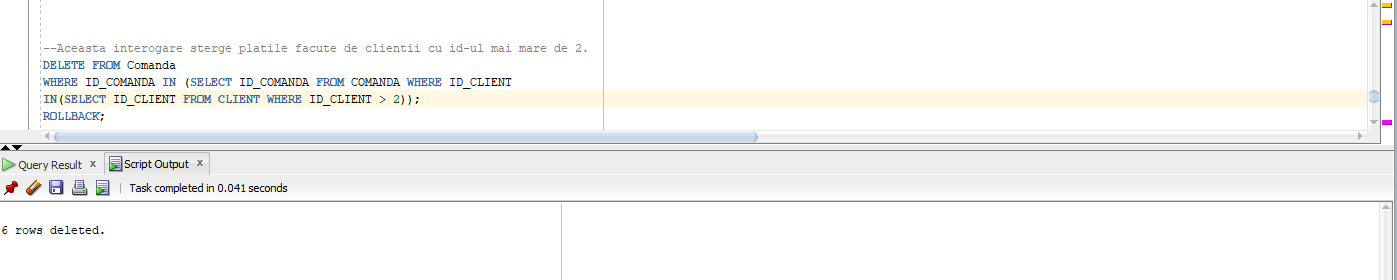


2. Sa se stearga platile facute de clientii cu id-ul mai mare de 2.

DELETE FROM Comanda

WHERE ID\_COMANDA IN (SELECT ID\_COMANDA FROM COMANDA WHERE ID\_CLIENT

IN(SELECT ID\_CLIENT FROM CLIENT WHERE ID\_CLIENT > 2));



3.Sa se mareasca salariul angajatilor de la Magazinul 2 cu 20% .

UPDATE Angajat

SET Salariu\_Angajat = Salariu\_Angajat \* 1.2

WHERE ID\_Angajat IN (

SELECT AM.ID\_Angajat

FROM Angajat\_Magazin AM

INNER JOIN Magazin M ON AM.ID\_Magazin = M.ID\_Magazin

WHERE M.Nume = 'Magazin2'

);

