

GESTIUNEA UNEI FLORARII

-Proiect baze de date-

INFORMATICA- Anul 1

Dogareci Bianca-Alexandra

Grupa 143

Cerinta 1:

1. Descrierea modelului real si a utilitatii acestuia:

Pentru acest proiect tema pe care am ales-o este o florarie care se ocupa cu vanzarea de produse online, adica vanzarea unor diferite tipuri de flori, a unor ingrasaminte pentru plante etc. In baza de date a florariei sunt stocate informatii esentiale, importante cu privire la produse, client, comenziile plasate de clienti, angajati s.a.m.d.

Utilitatea bazei de date consta in capacitatea de a gestiona informatiile, de a putea urmari si analiza comportamentul clientilor, de a gestiona stocurile, de a optimiza procesul de vanzare, de a putea vedea ce angajat se ocupa de o anumita comanda si multe altele.

2. Reguli de functionare:

- Angajatii pot ocupa un singur rol.
- Rulurile pot fi atribuite mai multor angajati.
- Un angajat poate fi responsabil de mai multe comenzi.
- O comanda este pregatita de un singur angajat.
- Un client poate pesta mai multe comenzi.
- O comanda poate corespunde unui singur client.
- O adresa poate corespunde mai multor clienti.
- Un client poate avea numai o adresa.
- Un client poate da mai multe recenzii.
- O recenzie poate fi a unui singur client.
- O comanda poate avea mai multe produse.
- Un produs poate apartine mai multor comenzi.
- Un furnizor poate aduce mai multe produse.
- Un produs poate fi adus numai de un furnizor.
- Un produs corespunde unei singure categorii de produse.
- O categorie de produse poate contine mai multe produse.

Cerinta 2:

Prezentarea constrangerilor (restrictii, reguli) impuse asupra modelului:

Acstea constrângeri sunt esențiale pentru a asigura integritatea datelor și relațiilor între entitățile noastre. Vom folosi diverse tipuri de constrângeri pentru a ne asigura că baza noastră de date funcționează într-un mod eficient și coerent.

1. Constrangeri de chei primare

Fiecare entitate din baza noastră de date are o cheie primară unică, care identifică în mod unic înregistrările din tabelă.

2. Constrangeri de chei externe

Implementăm constrângeri de chei externe pentru a asigura relații valide între entitățile noastre.

3. Constrangeri de tipul UNIQUE

Implementăm constrângeri de unicitate pentru a preveni duplicarea datelor în baza noastră de date. Ne ajuta să ne asigurăm că anumite campuri au valori unice în cadrul unor entități. Un exemplu il reprezintă numărul de telefon al fiecarui furzinor trebuie să fie unic, orice cheie primară etc.

4. Constrangeri de tipul NOT NULL

Cu aceasta constrângere ne asigurăm că anumite campuri nu pot să fie lăsate necomplete. Un exemplu il reprezintă numele rolului nu poate fi lăsat necompletat.

5. Constrangeri de tipul NULL

Cu aceasta constrângere ne asigurăm că anumite campuri pot să fie lăsate necomplete. Un exemplu il reprezintă descrierea rolului poate fi lăsată necompletată.

6. Constrangeri de tipul DEFAULT

Cu aceasta constrângere este specificată o valoare default pe care să o ia un anumit atribut în cazul în care aceasta nu este specificată. Un exemplu il reprezintă metoda de plată care este implicit "cash".

7. Constrangeri de tipul CHECK

Cu aceasta constrângere verificăm dacă o condiție logică este adevarată pentru valorile unui anumit atribut și permite inserarea sau actualizarea doar a valorilor care respectă această condiție. Un exemplu il reprezintă metoda de plată care poate lua valorile "cash" sau "card".

Cerinta 3:

Descrierea entitatilor, incluzand precizarea cheii primare:

1. Entitatea "ANGAJAT"

Entitatea "Angajat" reprezintă informațiile despre angajații companiei. Cheia primara este "id_angajat".

2. Entitatea "ROL ANGAJAT"

Entitatea "Rol angajat" descrie diferitele roluri pe care le poate avea un angajat în cadrul companiei. Cheia primara este "id_rol".

3. Entitatea "COMANDA"

Entitatea "Comanda" reprezintă informațiile despre comenziile plasate de clienți. Cheia primara este "id_comanda".

4. Entitatea "CLIENT"

Entitatea "Client" descrie informațiile despre clienții companiei. Cheia primara este "id_client".

5. Entitatea "ADRESA"

Entitatea "Adresa" conține informații despre adresele asociate clienților. Cheia primara este "id_adresa".

6. Entitatea "RECENZIE"

Entitatea "Recenzie" stochează recenzii ale clienților cu privire la produse sau servicii. Cheia primara este "id_recenzie".

7. Entitatea "PRODUS"

Entitatea "Produs" reprezintă informații despre produsele disponibile în cadrul companiei. Cheia primara este "id_produs".

8. Entitatea "FURNIZOR"

Entitatea "Furnizor" conține informații despre furnizorii produselor. Cheia primara este "id_furnizor".

9. Entitatea "CATEGORIE PRODUS"

Entitatea "Categorie produs" descrie diferitele categorii în care pot fi încadrate produsele. Cheia primara este "id_categorie_produs".

Cerinta 4:

Descrierea relațiilor, incluzând precizarea cardinalității acestora.

1. Relatia "ANGAJAT" – "ROL ANGAJAT":

Un angajat poate ocupa un singur rol (1) și un rol poate fi atribuit mai multor angajați (M). Aceasta relație este stabilită prin intermediul atributului id_rol, care se află în ambele entități.

2. Relația "ANGAJAT" - "COMANDA":

Un angajat poate fi responsabil de mai multe comenzi (M) și o comandă este pregătită de un singur angajat (1). Aceasta relație este stabilită prin intermediul atributului id_angajat, care se află în ambele entități.

3. Relația "CLIENT" - "COMANDA":

Un client poate plasa mai multe comenzi (M) și o comandă poate corespunde unui singur client (1). Aceasta relație este stabilită prin intermediul atributului id_client, care se află în ambele entități.

4. Relația "CLIENT" - "ADRESA":

O adresă poate corespunde mai multor clienți (M) și un client poate avea numai o adresă (1). Aceasta relație este stabilită prin intermediul atributului id_adresa, care se află în ambele entități.

5. Relația "CLIENT" - "RECENZIE":

Un client poate da mai multe recenzii (M) și o recenzie poate fi a unui singur client (1). Aceasta relație este stabilită prin intermediul atributului id_client, care se află în ambele entități.

6. Relația "COMANDA" - "PRODUS":

O comandă poate avea mai multe produse (M) și un produs poate apartine mai multor comenzi (M).

7. Relația "PRODUS" - "FURNIZOR":

Un furnizor poate aduce mai multe produse (M) și un produs poate fi adus numai de un furnizor (1). Aceasta relație este stabilită prin intermediul atributului id_furnizor, care se află în ambele entități.

8. Relația "PRODUS" - "CATEGORIE PRODUS":

Un produs corespunde unei singure categorii de produse (1) și o categorie de produse poate conține mai multe produse (M). Aceasta relație este stabilită prin intermediul atributului id_categprod, care se află în ambele entități.

Cerinta 5:

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

Entitatea “ANGAJAT”:

Atribut	Tip	Dimensiune	Valori posibile	Constrângeri
id_angajat	number	N/A	N/A	(PK) NOT NULL, UNIQUE
NumeAngajat	varchar2	N/A	N/A	NOT NULL
PrenumeAngajat	varchar2	N/A	N/A	NOT NULL
DataAngajare	date	N/A	N/A	NOT NULL
Salariu	number	N/A	N/A	NOT NULL
id_rol	number	N/A	N/A	(FK) NOT NULL

Entitatea “ROL ANGAJAT”:

Atribut	Tip	Dimensiune	Valori posibile	Constrângeri
id_rol	number	N/A	N/A	(PK) NOT NULL, UNIQUE
NumeRol	varchar2	N/A	N/A	NOT NULL
DescriereRol	varchar2	N/A	N/A	N/A

Entitatea “COMANDA”:

Atribut	Tip	Dimensiune	Valori posibile	Constrangeri
id_comanda	number	N/A	N/A	(PK) NOT NULL, UNIQUE
DataComanda	date	N/A	N/A	NOT NULL
TipPlata	varchar2	N/A	CHECK: cash, card DEFAULT: cash	NOT NULL
id_angajat	number	N/A	N/A	(FK) NOT NULL
id_client	number	N/A	N/A	(FK) NOT NULL

Entitatea “CLIENT”:

Atribut	Tip	Dimensiune	Valori posibile	Constrangeri
id_client	number	N/A	N/A	(PK) NOT NULL, UNIQUE
NumeClient	varchar2	N/A	N/A	NOT NULL
PrenumeClient	varchar2	N/A	N/A	NOT NULL
EmailClient	varchar2	N/A	N/A	NOT NULL, UNIQUE
TelefonClient	varchar2	10	N/A	NOT NULL, UNIQUE
id_adresa	number	N/A	N/A	(FK) NOT NULL

Entitatea "ADRESA":

Atribut	Tip	Dimensiune	Valori posibile	Constrangeri
id_adresa	number	N/A	N/A	(PK) NOT NULL, UNIQUE
Oras	varchar2	N/A	N/A	NOT NULL
Strada	varchar2	N/A	N/A	NOT NULL
Numar	number	N/A	N/A	NOT NULL
Bloc	varchar2	N/A	N/A	NULL
Scara	number	N/A	N/A	NULL
Etaj	number	N/A	N/A	NULL
Apartament	number	N/A	N/A	NULL
CodPostal	varchar2	N/A	N/A	NOT NULL

Entitatea "RECENZIE":

Atribut	Tip	Dimensiune	Valori posibile	Constrangeri
id_recenzie	number	N/A	N/A	(PK) NOT NULL, UNIQUE
Evaluare	number	N/A	N/A	NOT NULL
Comentariu	varchar2	N/A	N/A	NULL
id_client	number	N/A	N/A	(FK) NOT NULL

Entitatea “PRODUS”:

Atribut	Tip	Dimensiune	Valori posibile	Constrangeri
id_produs	number	N/A	N/A	(PK) NOT NULL, UNIQUE
NumeProdus	varchar2	N/A	N/A	NOT NULL
Pret	number	N/A	N/A	NOT NULL
id_categprod	number	N/A	N/A	(FK) NOT NULL
id_furnizor	number	N/A	N/A	(FK) NOT NULL

Entitatea “FURNIZOR”:

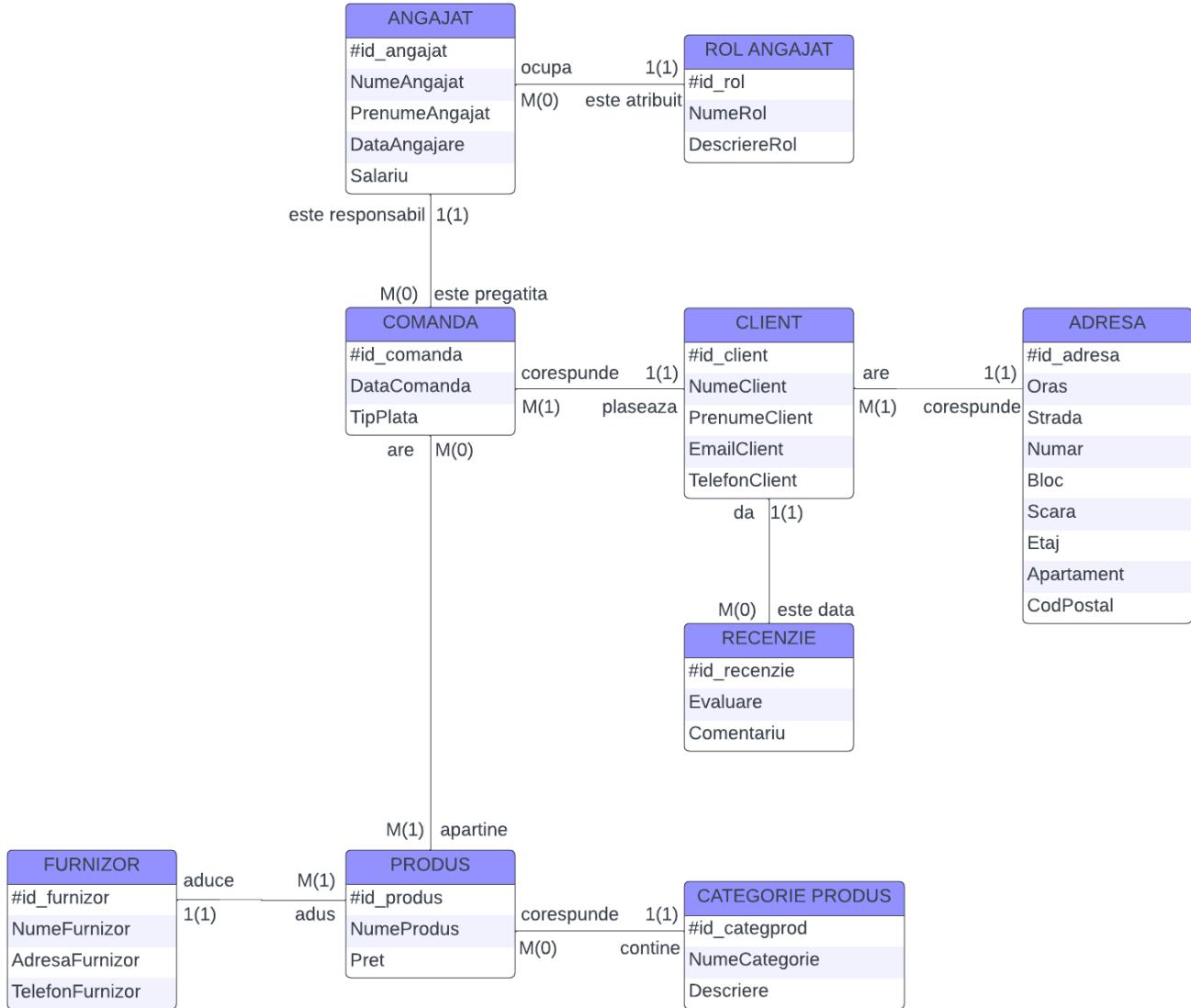
Atribut	Tip	Dimensiune	Valori posibile	Constrangeri
id_furnizor	number	N/A	N/A	(PK) NOT NULL, UNIQUE
NumeFurnizor	varchar2	N/A	N/A	NOT NULL
AdresaFurnizor	varchar2	N/A	N/A	NOT NULL, UNIQUE
TelefonFurnizor	varchar2	10	N/A	NOT NULL, UNIQUE

Entitatea “CATEGORIE PRODUS”:

Atribut	Tip	Dimensiune	Valori posibile	Constrangeri
id_categprod	number	N/A	N/A	(PK) NOT NULL, UNIQUE
NumeCategoria	varchar2	N/A	N/A	NOT NULL
Descriere	varchar2	N/A	N/A	NULL

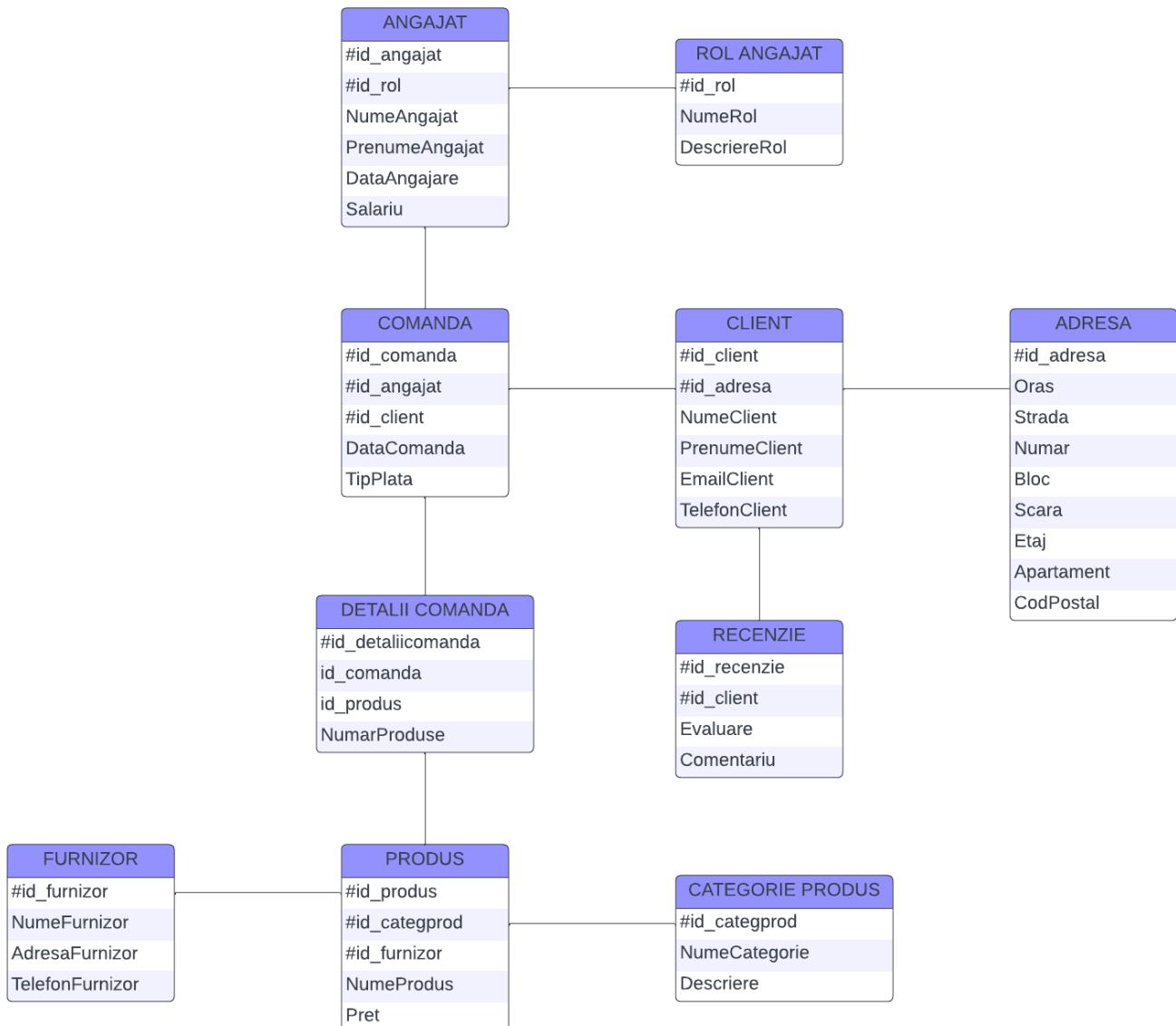
Cerinta 6:

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



Cerinta 7:

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.



Cerinta 8:

Enumerarea schemelor relaționale corespunzătoare diagramei conceptuale proiectate la punctul 7.

1. ANGAJAT (#id_angajat, NumeAngajat, PrenumeAngajat, DataAngajare, Salariu, id_rol)
2. ROL ANGAJAT (#id_rol, NumeRol, DescriereRol)
3. COMANDA (#id_comanda, DataComanda, TipPlata, id_angajat, id_client)
4. CLIENT (#id_client, NumeClient, PrenumeClient, EmailClient, TelefonClient, id_adresa)
5. ADRESA (#id_adresa, Oras, Strada, Numar, Bloc, Scara, Etaj, Apartament, CodPostal)
6. RECENZIE (#id_recenzie, Evaluare, Comentariu, id_client)
7. DETALII COMANDA (#id_detaliicomanda, id_produs, id_comanda, NumarProduse)
8. PRODUS (#id_produs, NumeProdus, Pret, id_categprod, id_furnizor)
9. FURNIZOR (#id_furnizor, NumeFurnizor, AdresaFurnizor, TelefonFurnizor)
10. CATEGORIE PRODUS (#id_categprod, NumeCategorie, Descriere)

Cerinta 9:

Realizarea normalizării pana la forma normala 3(FN1-FN3).

- Forma normala 1 (FN1)

O relatie se considera in prima forma normala atunci cand fiecare inregistrare este unic identificata prin intermediul unei chei primare si fiecare atribut contine doar o valoare singula. In aceasta forma normala, nu sunt premise grupuri repetitive de atribute.

Luam ca exemplu entitatile FURNIZOR si PRODUS.

1) Mai multe valori in acelasi atribut

FURNIZOR	PRODUS
Fresco Verde	cherry brandy, cosima, abstract
Happy Flower Holland	royal red, empire blue

Interogarile pentru a selecta inregistrari cand un atribut contine mai multe valori semnificative sunt foarte dificile.

FURNIZOR	PRODUS
Fresco Verde	cherry brandy, cosima, abstract
Happy Flower Holland	royal red, empire blue



FURNIZOR	PRODUS
Fresco Verde	cherry brandy
Fresco Verde	cosima
Fresco Verde	abstract
Happy Flower Holland	royal red
Happy Flower Holland	empire blue



id	FURNIZOR	PRODUS
1	Fresco Verde	cherry brandy
2	Fresco Verde	cosima
3	Fresco Verde	abstract
4	Happy Flower Holland	royal red
5	Happy Flower Holland	empire blue

2) Mai multe coloane contin acelasi tip de date

FURNIZOR	PRODUS_1	PRODUS_2	PRODUS_3
Fresco Verde	cherry brandy	cosima	abstract
Happy Flower Holland	royal red	empire blue	



FURNIZOR	PRODUS
Fresco Verde	cherry brandy
Fresco Verde	cosima
Fresco Verde	abstract
Happy Flower Holland	royal red
Happy Flower Holland	empire blue



id	FURNIZOR	PRODUS
1	Fresco Verde	cherry brandy
2	Fresco Verde	cosima
3	Fresco Verde	abstract
4	Happy Flower Holland	royal red
5	Happy Flower Holland	empire blue

- Forma normala 2 (FN2)

Daca unul sau mai multe coloane sunt dependente numai de un id, atunci ele trebuie sa faca parte dintr-un alt tabel.

Luam ca exemplu entitatile ANGAJAT si ROL ANGAJAT.

id_angajat	NumeAngajat	PrenumeAngajat	DataAngajare	Salariu	id_rol	NumeRol	DescriereRol
1	Popa	Stefan	18-08-2019	4000	10	Florar	Angajatul responsabil cu aranjarea buchetelor.
2	Ionescu	Anastasia	20-03-2020	5000	10	Florar	Angajatul responsabil cu aranjarea buchetelor.



id_angajat	NumeAngajat	PrenumeAngajat	DataAngajare	Salariu	id_rol
1	Popa	Stefan	18-08-2019	4000	10
2	Ionescu	Anastasia	20-03-2020	5000	10

id_rol	NumeRol	DescriereRol
10	Florar	Angajatul responsabil cu aranjarea buchetelor.
10	Florar	Angajatul responsabil cu aranjarea buchetelor.

Atributele cu numele NumeRol si DescriereRol depind doar de cheia id_rol, nu si de id_angajat, asa ca despartim datele in doua tabele ca mai sus.

- Forma normala 3 (FN3)

O relatie se considera in forma normala trei daca:

- se gaseste in FN2
- fiecare atribut care nu este cheie primara depinde direct de aceasta

Luam ca exemplu entitatile ANGAJAT si ROL ANGAJAT.

id_angajat	NumeAngajat	PrenumeAngajat	DataAngajare	Salariu	NumarRol	NumeRol	DescriereRol
1	Popa	Stefan	18-08-2019	4000	10	Florar	Angajatul responsabil cu aranjarea buchetelor.
2	Ionescu	Anastasia	20-03-2020	5000	10	Florar	Angajatul responsabil cu aranjarea buchetelor.



id_angajat	NumeAngajat	PrenumeAngajat	DataAngajare	Salariu	id_rol
1	Popa	Stefan	18-08-2019	4000	10
2	Ionescu	Anastasia	20-03-2020	5000	10

id_rol	NumeRol	DescriereRol
10	Florar	Angajatul responsabil cu aranjarea buchetelor.
10	Florar	Angajatul responsabil cu aranjarea buchetelor.

Observam ca atributele NumeRol si DescriereRol nu depind de cheia primara id_angajat, ci de atributul NumarRol, asa ca le scoatem din tabelul ANGAJAT si cream tabelul ROL ANGAJAT. Astfel atributul NumeRol devine id_rol, primary key in tabelul ROL ANGAJAT si foreign key in tabelul ANGAJAT.

Cerinta 10:

Crearea unei secvențe ce va fi utilizată în inserarea înregistrărilor în tabele (punctul 11).

Secventa cu care adaugam automat id-uri unice in tabele:

```
create sequence flor_seq
```

```
    start with 1
```

```
    increment by 1
```

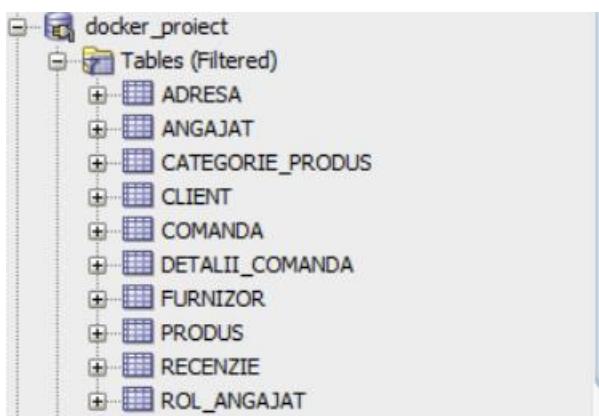
```
    nocache
```

```
    nocycle;
```

```
create sequence flor_seq
    start with 1
    increment by 1
    nocache
    nocycle;
```

Cerinta 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).



- **TABELA ROL_ANGAJAT:**

```
create table ROL_ANGAJAT  
  (id_rol number(5) constraint pkey_rol primary key,  
   NumeRol varchar(20) constraint NumeRol not null,  
   DescriereRol varchar(100) constraint DescriereRol null  
);
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_ROL	NUMBER (5, 0)	No	(null)	1	(null)
2 NUMEROL	VARCHAR2 (20 BYTE)	No	(null)	2	(null)
3 DESCRIEREROL	VARCHAR2 (100 BYTE)	Yes	(null)	3	(null)

```
insert into ROL_ANGAJAT  
values (flor_seq.nextval,'Florar','Angajatul responsabil cu aranjarea buchetelor  
și compozițiilor florale.');
```

```
insert into ROL_ANGAJAT  
values (flor_seq.nextval,'Manager','Angajatul cu rol de supervizare și  
coordonare a operațiunilor zilnice ale florăriei.');
```

```
insert into ROL_ANGAJAT  
values (flor_seq.nextval,'Specialist design','Angajatul cu expertiză în creația de  
aranjamente florale pentru evenimente.');
```

insert into ROL_ANGAJAT

values (flor_seq.nextval,'Vânzător','Angajatul care interacționează direct cu clienții.');

insert into ROL_ANGAJAT

values (flor_seq.nextval,'Livrator','Angajatul responsabil cu livrarea florilor și aranjamentelor la adresele indicate de clienți.');

ID_ROL	NUMEROL	DESCRIEREROL
1	Florar	Angajatul responsabil cu aranjarea buchetelor și componzițiilor florale.
2	Manager	Angajatul cu rol de supervizare și coordonare a operațiunilor zilnice ale florăriei.
3	Specialist design	Angajatul cu expertiză în creația de aranjamente florale pentru evenimente.
4	Vânzător	Angajatul care interacționează direct cu clienții.
5	Livrator	Angajatul responsabil cu livrarea florilor și aranjamentelor la adresele indicate de clienți.

Codul:

```
create table ROL_ANGAJAT
(id_rol number(5) constraint pkey_rol primary key,
NumeRol varchar(20) constraint NumeRol not null,
DescriereRol varchar(100) constraint DescriereRol null
);

insert into ROL_ANGAJAT
values(flor_seq.nextval,'Florar','Angajatul responsabil cu aranjarea buchetelor și componzițiilor florale.');

insert into ROL_ANGAJAT
values(flor_seq.nextval,'Manager','Angajatul cu rol de supervizare și coordonare a operațiunilor zilnice ale florăriei.');

insert into ROL_ANGAJAT
values(flor_seq.nextval,'Specialist design','Angajatul cu expertiză în creația de aranjamente florale pentru evenimente.');

insert into ROL_ANGAJAT
values(flor_seq.nextval,'Vânzător','Angajatul care interacționează direct cu clienții.');

insert into ROL_ANGAJAT
values(flor_seq.nextval,'Livrator','Angajatul responsabil cu livrarea florilor și aranjamentelor la adresele indicate de clienți.');
```

- **TABELA ANGAJAT:**

```
create table ANGAJAT
```

```
    (id_angajat number(5) constraint pkey_angajat primary key,
```

```
        id_rol number(5), constraint fk_angajat_rol foreign key(id_rol) references
ROL_ANGAJAT(id_rol),
```

```
        NumeAngajat varchar(30) constraint NumeAngajat not null,
```

```
        PrenumeAngajat varchar(30) constraint PrenumeAngajat not null,
```

```
        DataAngajare date constraint DataAngajare not null,
```

```
        Salariu number(10) constraint Salariu not null
```

```
);
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_ANGAJAT	NUMBER (5, 0)	No	(null)	1	(null)
2 ID_ROL	NUMBER (5, 0)	Yes	(null)	2	(null)
3 NUMEANGAJAT	VARCHAR2 (30 BYTE)	No	(null)	3	(null)
4 PRENUMEANGAJAT	VARCHAR2 (30 BYTE)	No	(null)	4	(null)
5 DATAANGAJARE	DATE	No	(null)	5	(null)
6 SALARIU	NUMBER (10, 0)	No	(null)	6	(null)

```
insert into ANGAJAT
```

```
values(flor_seq.nextval,(select id_rol from ROL_ANGAJAT where
NumeRol='Florar'),'Ionescu','Anastasia',to_date('16-08-2019','DD-MM-
YYYY'),5000);
```

```

insert into ANGAJAT
values(flor_seq.nextval,(select id_rol from ROL_ANGAJAT where
NumeRol='Florar'),'Florescu','Letitia',to_date('04-07-2019','DD-MM-
YYYY'),5100);

insert into ANGAJAT
values(flor_seq.nextval,(select id_rol from ROL_ANGAJAT where
NumeRol='Manager'),'Stanescu','Stefan',to_date('01-12-2005','DD-MM-
YYYY'),9000);

insert into ANGAJAT
values(flor_seq.nextval,(select id_rol from ROL_ANGAJAT where
NumeRol='Specialist design'),'Dumitrache','Rafael',to_date('10-03-2019','DD-
MM-YYYY'),6200);

insert into ANGAJAT
values(flor_seq.nextval,(select id_rol from ROL_ANGAJAT where
NumeRol='Specialist design'),'Teodorescu','Eleonora',to_date('15-04-
2015','DD-MM-YYYY'),6400);

```

ID_ANGAJAT	ID_ROL	NUMEANGAJAT	PRENUMEANGAJAT	DATAANGAJARE	SALARIU
1	6	1 Ionescu	Anastasia	16-AUG-19	5000
2	7	1 Florescu	Letitia	04-JUL-19	5100
3	8	2 Stanescu	Stefan	01-DEC-05	9000
4	9	3 Dumitrache	Rafael	10-MAR-19	6200
5	10	3 Teodorescu	Eleonora	15-APR-15	6400

Codul:

```
create table ANGAJAT
(id_angajat number(5) constraint pkey_angajat primary key,
id_rol number(5), constraint fk_angajat_rol foreign key(id_rol) references ROL_ANGAJAT(id_rol),
NumeAngajat varchar(30) constraint NumeAngajat not null,
PrenumeAngajat varchar(30) constraint PrenumeAngajat not null,
DataAngajare date constraint DataAngajare not null,
Salariu number(10) constraint Salariu not null
);

insert into ANGAJAT
values(flor_seq.nextval,(select id_rol from ROL_ANGAJAT where NumeRol='Florar'),'Ionescu','Anastasia',to_date('16-08-2019','DD-MM-YYYY'),5000);

insert into ANGAJAT
values(flor_seq.nextval,(select id_rol from ROL_ANGAJAT where NumeRol='Florar'),'Florescu','Letitia',to_date('04-07-2019','DD-MM-YYYY'),5100);

insert into ANGAJAT
values(flor_seq.nextval,(select id_rol from ROL_ANGAJAT where NumeRol='Manager'),'Stanescu','Stefan',to_date('01-12-2005','DD-MM-YYYY'),9000);

insert into ANGAJAT
values(flor_seq.nextval,(select id_rol from ROL_ANGAJAT where NumeRol='Specialist design'),'Dumitache','Rafael',to_date('10-03-2019','DD-MM-YYYY'),6200);

insert into ANGAJAT
values(flor_seq.nextval,(select id_rol from ROL_ANGAJAT where NumeRol='Specialist design'),'Teodorescu','Eleonora',to_date('15-04-2015','DD-MM-YYYY'),6400);
```

• TABELA ADRESA:

create table ADRESA

```
(id_adresa number(5) constraint pkey_adresa primary key,
Oras varchar(20) constraint Oras not null,
Strada varchar(30) constraint Strada not null,
Numar number(5) constraint Numar not null,
Bloc varchar(20) constraint Bloc null,
Scara number(5) constraint Scara null,
Etaj number(5) constraint Etaj null,
Apartament number(5) constraint Apartament null,
CodPostal varchar(20) constraint CodPostal not null
);
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_ADRESA	NUMBER(5,0)	No	(null)	1	(null)
2 ORAS	VARCHAR2(20 BYTE)	No	(null)	2	(null)
3 STRADA	VARCHAR2(30 BYTE)	No	(null)	3	(null)
4 NUMAR	NUMBER(5,0)	No	(null)	4	(null)
5 BLOC	VARCHAR2(20 BYTE)	Yes	(null)	5	(null)
6 SCARA	NUMBER(5,0)	Yes	(null)	6	(null)
7 ETAJ	NUMBER(5,0)	Yes	(null)	7	(null)
8 APARTAMENT	NUMBER(5,0)	Yes	(null)	8	(null)
9 CODPOSTAL	VARCHAR2(20 BYTE)	No	(null)	9	(null)

insert into ADRESA

```
values (flor_seq.nextval, 'Bucuresti', 'Str. Victoriei', 10, null, null, null, null, '010101');
```

insert into ADRESA

```
values (flor_seq.nextval, 'Cluj-Napoca', 'Str. Bucuriei', 15, 'A', 2, 3, 7, '400000');
```

insert into ADRESA

```
values (flor_seq.nextval, 'Timisoara', 'Str. Mihai Viteazu', 8, 'B3', 1, 6, 60, '300100');
```

insert into ADRESA

```
values (flor_seq.nextval, 'Iasi', 'Str. Stefan Cel Mare', 25, null, null, null, null, '700300');
```

insert into ADRESA

```
values (flor_seq.nextval, 'Constanta', 'Bd. Mamaia', 20, null, null, null, null, '900001');
```

	ID_ADRESA	ORAS	STRADA	NUMAR	BLOC	SCARA	ETAJ	APARTAMENT	CODPOSTAL
1	11	Bucuresti	Str. Victoriei	10	(null)	(null)	(null)	(null)	010101
2	12	Cluj-Napoca	Str. Bucuriei	15	A		2	3	7 400000
3	13	Timisoara	Str. Mihai Viteazu	8	B3		1	6	60 300100
4	14	Iasi	Str. Stefan Cel Mare	25	(null)	(null)	(null)	(null)	700300
5	15	Constanta	Bd. Mamaia	20	(null)	(null)	(null)	(null)	900001

Codul:

```
create table ADRESA
(id_adresa number(5) constraint pkey_adresa primary key,
Oras varchar(20) constraint Oras not null,
Strada varchar(30) constraint Strada not null,
Numar number(5) constraint Numar not null,
Bloc varchar(20) constraint Bloc null,
Scara number(5) constraint Scara null,
Etaj number(5) constraint Etaj null,
Apartament number(5) constraint Apartament null,
CodPostal varchar(20) constraint CodPostal not null
);

insert into ADRESA
values (flor_seq.nextval, 'Bucuresti', 'Str. Victoriei', 10, null, null, null, null, '010101');

insert into ADRESA
values (flor_seq.nextval, 'Cluj-Napoca', 'Str. Bucuriei', 15, 'A', 2, 3, 7, '400000');

insert into ADRESA
values (flor_seq.nextval, 'Timisoara', 'Str. Mihai Viteazu', 8, 'B3', 1, 6, 60, '300100');

insert into ADRESA
values (flor_seq.nextval, 'Iasi', 'Str. Stefan Cel Mare', 25, null, null, null, null, '700300');

insert into ADRESA
values (flor_seq.nextval, 'Constanta', 'Bd. Mamaia', 20, null, null, null, null, '900001');
```

- **TABELA CLIENT:**

create table CLIENT

```
(id_client number(5) constraint pkey_client primary key,
 id_adresa number(5), constraint fk_client_adresa foreign key(id_adresa)
 references ADRESA(id_adresa),
 NumeClient varchar(30) constraint NumeClient not null,
 PrenumeClient varchar(30) constraint PrenumeClient not null,
 EmailClient varchar(30) constraint EmailClient not null unique,
 TelefonClient varchar(30) constraint TelefonClient not null unique
);
```

The screenshot shows the 'Columns' tab for the 'CLIENT' table in Oracle SQL Developer. The table structure is as follows:

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_CLIENT	NUMBER (5, 0)	No	(null)	1	(null)
2 ID_ADRESA	NUMBER (5, 0)	Yes	(null)	2	(null)
3 NUMECLIENT	VARCHAR2 (30 BYTE)	No	(null)	3	(null)
4 PRENUMECCLIENT	VARCHAR2 (30 BYTE)	No	(null)	4	(null)
5 EMAILCLIENT	VARCHAR2 (30 BYTE)	No	(null)	5	(null)
6 TELEFONCLIENT	VARCHAR2 (30 BYTE)	No	(null)	6	(null)

insert into CLIENT

values

```
(flor_seq.nextval,11,'Popescu','Ioana','ioanapopescu@gmail.com','074432218
8');
```

```
insert into CLIENT  
values  
(flor_seq.nextval,12,'Apostol','Teodora','teodoraapostol@yahoo.com','074432  
5438');
```

```
insert into CLIENT  
values  
(flor_seq.nextval,13,'Olteanu','Stefan','stefanolteanu@gmail.com','073452218  
8');
```

```
insert into CLIENT  
values  
(flor_seq.nextval,14,'Tintea','Alexandru','alextintea@gmail.com','0744678388'  
)
```

```
insert into CLIENT  
values (flor_seq.nextval,15,'Nica','Ana','ananica@ymail.com','0722722722');
```

```
insert into CLIENT  
values  
(flor_seq.nextval,11,'Feurdean','Mihai','mihaifeur@gmail.com','0747777788');
```

ID_CLIENT	ID_ADRESA	NUMECLIENT	PRENUMECLIENT	EMAILCLIENT	TELEFONCLIENT
1	16	11 Popescu	Ioana	ioanapopescu@gmail.com	0744322188
2	17	12 Apostol	Teodora	teodoraapostol@yahoo.com	0744325438
3	18	13 Olteanu	Stefan	stefanolteanu@gmail.com	0734522188
4	19	14 Tintea	Alexandru	alextintea@gmail.com	0744678388
5	20	15 Nica	Ana	ananica@ymail.com	0722722722
6	21	11 Feurdean	Mihai	mihaifeur@gmail.com	0747777788

Codul:

```
create table CLIENT
(id_client number(5) constraint pkey_client primary key,
id_adresa number(5), constraint fk_client_adresa foreign key(id_adresa) references ADRESA(id_adresa),
NumeClient varchar(30) constraint NumeClient not null,
PrenumeClient varchar(30) constraint PrenumeClient not null,
EmailClient varchar(30) constraint EmailClient not null unique,
TelefonClient varchar(30) constraint TelefonClient not null unique
);

insert into CLIENT
values (flor_seq.nextval,11,'Popescu','Ioana','ioanapopescu@gmail.com','0744322188');

insert into CLIENT
values (flor_seq.nextval,12,'Apostol','Teodora','teodoraapostol@yahoo.com','0744325438');

insert into CLIENT
values (flor_seq.nextval,13,'Olteanu','Stefan','stefanolteanu@gmail.com','0734522188');

insert into CLIENT
values (flor_seq.nextval,14,'Tintea','Alexandru','alextintea@gmail.com','0744678388');

insert into CLIENT
values (flor_seq.nextval,15,'Nica','Ana','ananica@ymail.com','0722722722');

insert into CLIENT
values (flor_seq.nextval,11,'Feurdean','Mihai','mihafeur@gmail.com','0747777788');
```

• TABELA RECENZIE:

```
create table RECENZIE
(id_recenzie number(5) constraint pkey_recenzie primary key,
id_client number(5), constraint fk_recenzie_client foreign key(id_client)
references CLIENT(id_client),
Evaluare number(2) constraint Evaluare not null,
Comentariu varchar(100) constraint Comentariu null
);
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_RECENZIE	NUMBER (5,0)	No	(null)	1	(null)
2 ID_CLIENT	NUMBER (5,0)	Yes	(null)	2	(null)
3 EVALUARE	NUMBER (2,0)	No	(null)	3	(null)
4 COMENTARIU	VARCHAR2 (100 BYTE)	Yes	(null)	4	(null)

insert into RECENZIE

```
values (flor_seq.nextval,(select id_client from CLIENT where
EmailClient='mihaifeur@gmail.com'),10,'Foarte mulțumit de servicii, personal
amabil.');
```

insert into RECENZIE

```
values (flor_seq.nextval,(select id_client from CLIENT where
EmailClient='mihaifeur@gmail.com'),9,null);
```

insert into RECENZIE

```
values (flor_seq.nextval,(select id_client from CLIENT where
EmailClient='ananica@ymail.com'),8,'Produse de calitate, dar livrarea a
întârziat.');
```

insert into RECENZIE

```
values (flor_seq.nextval,(select id_client from CLIENT where
EmailClient='ioanapopescu@gmail.com'),5,'Nu am fost impresionata de
calitatea produselor.');
```

insert into RECENZIE
values (flor_seq.nextval,(select id_client from CLIENT where EmailClient='alextintea@gmail.com'),10,'Recomand cu încredere, livrare rapidă și produse proaspete.');

	ID_RECENZIE	ID_CLIENT	EVALUARE	COMENTARIU
1	22	21	10	Foarte mulțumit de servicii, personal amabil.
2	23	21	9	(null)
3	24	20	8	Produse de calitate, dar livrarea a întârziat.
4	25	16	5	Nu am fost impresionata de calitatea produselor.
5	26	19	10	Recomand cu încredere, livrare rapidă și produse proaspete.

Codul:

```
create table RECENZIE
(id_recenzie number(5) constraint pkey_recenzie primary key,
id_client number(5), constraint fk_recenzie_client foreign key(id_client) references CLIENT(id_client),
Evaluare number(2) constraint Evaluare not null,
Comentariu varchar(100) constraint Comentariu null
);

insert into RECENZIE
values (flor_seq.nextval,(select id_client from CLIENT where EmailClient='mihaiifeur@gmail.com'),10,'Foarte mulțumit de servicii, personal amabil.');

insert into RECENZIE
values (flor_seq.nextval,(select id_client from CLIENT where EmailClient='mihaiifeur@gmail.com'),9,null);

insert into RECENZIE
values (flor_seq.nextval,(select id_client from CLIENT where EmailClient='ananica@ymail.com'),8,'Produse de calitate, dar livrarea a întârziat.);

insert into RECENZIE
values (flor_seq.nextval,(select id_client from CLIENT where EmailClient='ioanapopescu@gmail.com'),5,'Nu am fost impresionata de calitatea produselor.);

insert into RECENZIE
values (flor_seq.nextval,(select id_client from CLIENT where EmailClient='alextintea@gmail.com'),10,'Recomand cu încredere, livrare rapidă și produse proaspete.');
```

- **TABELA COMANDA:**

create table COMANDA

```
(id_comanda number(5) constraint pkey_comanda primary key,
 id_angajat number(5), constraint fk_comanda_angajat foreign
 key(id_angajat) references ANGAJAT(id_angajat),
 id_client number(5), constraint fk_comanda_client foreign key(id_client)
 references CLIENT(id_client),
 DataComanda date constraint DataComanda not null,
 TipPlata varchar(10) default 'cash' constraint TipPlata not null
 );
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_COMANDA	NUMBER(5, 0)	No	(null)	1	(null)
2 ID_ANAJAT	NUMBER(5, 0)	Yes	(null)	2	(null)
3 ID_CLIENT	NUMBER(5, 0)	Yes	(null)	3	(null)
4 DATACOMANDA	DATE	No	(null)	4	(null)
5 TIPPLATA	VARCHAR2(10 BYTE)	No	'cash'	5	(null)

insert into COMANDA

```
values (flor_seq.nextval,6,(select id_client from CLIENT where
EmailClient='mihaifeur@gmail.com'),to_date('15-12-2020','DD-MM-
YYYY'),'card');
```

```

insert into COMANDA
values (flor_seq.nextval,6,(select id_client from CLIENT where
EmailClient='ananica@ymail.com'),to_date('05-05-2020','DD-MM-
YYYY'),'card');

insert into COMANDA
values (flor_seq.nextval,7,(select id_client from CLIENT where
EmailClient='ananica@ymail.com'),to_date('27-11-2020','DD-MM-
YYYY'),'cash');

insert into COMANDA
values (flor_seq.nextval,10,(select id_client from CLIENT where
EmailClient='ioanapopescu@gmail.com'),to_date('04-06-2020','DD-MM-
YYYY'),default);

insert into COMANDA
values (flor_seq.nextval,9,(select id_client from CLIENT where
EmailClient='alextintea@gmail.com'),to_date('04-06-2021','DD-MM-
YYYY'),'cash');

```

ID_COMANDA	ID_ANGAJAT	ID_CLIENT	DATA COMANDA	TIPPLATA
1	27	6	21 15-DEC-20	card
2	28	6	20 05-MAY-20	card
3	29	7	20 27-NOV-20	cash
4	30	10	16 04-JUN-20	cash
5	31	9	19 04-JUN-21	cash

Codul:

```
create table COMANDA
(id_comanda number(5) constraint pkey_comanda primary key,
id_angajat number(5), constraint fk_comanda_angajat foreign key(id_angajat) references ANGAJAT(id_angajat),
id_client number(5), constraint fk_comanda_client foreign key(id_client) references CLIENT(id_client),
DataComanda date constraint DataComanda not null,
TipPlata varchar(10) default 'cash' constraint TipPlata not null
);

insert into COMANDA
values (flor_seq.nextval,6,(select id_client from CLIENT where EmailClient='mihaiifeur@gmail.com'),to_date('15-12-2020','DD-MM-YYYY'),'card');

insert into COMANDA
values (flor_seq.nextval,6,(select id_client from CLIENT where EmailClient='ananica@ymail.com'),to_date('05-05-2020','DD-MM-YYYY'),'card');

insert into COMANDA
values (flor_seq.nextval,7,(select id_client from CLIENT where EmailClient='ananica@ymail.com'),to_date('27-11-2020','DD-MM-YYYY'),'cash');

insert into COMANDA
values (flor_seq.nextval,10,(select id_client from CLIENT where EmailClient='ioanapopescu@gmail.com'),to_date('04-06-2020','DD-MM-YYYY'),default);

insert into COMANDA
values (flor_seq.nextval,9,(select id_client from CLIENT where EmailClient='alextintea@gmail.com'),to_date('04-06-2021','DD-MM-YYYY'),'cash');
```

- **TABELA CATEGORIE_PRODUS:**

create table CATEGORIE_PRODUS

(id_categprod number(5) constraint pkey_categprod primary key,
NumeCategoria varchar(20) constraint NumeCategoria not null,
Descriere varchar(100) constraint Descriere null

);

The screenshot shows the Oracle SQL Developer interface with the 'CATEGORIE_PRODUS' table selected. The table has three columns: ID_CATEGPROD (NUMBER(5,0)), NUMECATEGORIE (VARCHAR2(20 BYTE)), and DESCRIERE (VARCHAR2(100 BYTE)). The primary key is defined on the first column.

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_CATEGPROD	NUMBER(5,0)	No	(null)	1	(null)
2 NUMECATEGORIE	VARCHAR2(20 BYTE)	No	(null)	2	(null)
3 DESCRIERE	VARCHAR2(100 BYTE)	Yes	(null)	3	(null)

insert into CATEGORIE_PRODUS

values (flor_seq.nextval,'Buchete','Buchete de flori proaspete și aranjamente florale elegante.');

```

insert into CATEGORIE_PRODUS
values (flor_seq.nextval,'Plante de interior',null);

```

```

insert into CATEGORIE_PRODUS
values (flor_seq.nextval,'Plante de exterior','Plante de exterior perene si
anuale');

```

```

insert into CATEGORIE_PRODUS
values (flor_seq.nextval,'Accesorii','Ghivece, ambalaje, felicitari, ingrasaminte,
decoratiuni etc.');

```

```

insert into CATEGORIE_PRODUS
values (flor_seq.nextval,'La bucata','Vanzare la fir.');

```

ID_CATEGPROD	NUMECATEGORIE	DESCRIERE
1	32 Buchete	Buchete de flori proaspete și aranjamente florale elegante.
2	33 Plante de interior	(null)
3	34 Plante de exterior	Plante de exterior perene și anuale
4	35 Accesorii	Ghivece, ambalaje, felicitari, ingrasaminte, decoratiuni etc.
5	36 La bucata	Vanzare la fir.

Codul:

```

create table CATEGORIE_PRODUS
(id_categprod number(5) constraint pkey_categprod primary key,
NumeCategoria varchar(20) constraint NumeCategoria not null,
Descriere varchar(100) constraint Descriere null
);

insert into CATEGORIE_PRODUS
values (flor_seq.nextval,'Buchete','Buchete de flori proaspete și aranjamente florale elegante.');

insert into CATEGORIE_PRODUS
values (flor_seq.nextval,'Plante de interior',null);

insert into CATEGORIE_PRODUS
values (flor_seq.nextval,'Plante de exterior','Plante de exterior perene și anuale');

insert into CATEGORIE_PRODUS
values (flor_seq.nextval,'Accesorii','Ghivece, ambalaje, felicitari, ingrasaminte, decoratiuni etc.');

insert into CATEGORIE_PRODUS
values (flor_seq.nextval,'La bucata','Vanzare la fir.');

```

- **TABELA FURNIZOR:**

```
create table FURNIZOR
```

```
(id_furnizor number(5) constraint pkey_furnizor primary key,  
NumeFurnizor varchar(30) constraint NumeFurnizor not null,  
AdresaFurnizor varchar(100) constraint AdresaFurnizor not null unique,  
TelefonFurnizor varchar(10) constraint TelefonFurnizor not null unique  
);
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_FURNIZOR	NUMBER(5,0)	No	(null)	1 (null)	
2 NUMEFURNIZOR	VARCHAR2(30 BYTE)	No	(null)	2 (null)	
3 ADRESAFURNIZOR	VARCHAR2(100 BYTE)	No	(null)	3 (null)	
4 TELEFONFURNIZOR	VARCHAR2(10 BYTE)	No	(null)	4 (null)	

```
insert into FURNIZOR
```

```
values (flor_seq.nextval,'Fresco Verde','Strada Trandafirilor 10, Bucureşti,  
010101, România','0721123456');
```

```
insert into FURNIZOR
```

```
values (flor_seq.nextval,'Happy Flower Holland','Aleea Florilor 5, Bucureşti,  
400000, România','0765987654');
```

```
insert into FURNIZOR
```

```
values (flor_seq.nextval,'GreenLeaf Plants','Bulevardul Liliacului 25, Bucureşti,  
700000, România','0744555888');
```

insert into FURNIZOR

values (flor_seq.nextval,'Natures Garden Supplies','Strada Irisului 8, Bucureşti, 300001, România','0782111222');

insert into FURNIZOR

values (flor_seq.nextval,'Bloom Nursery Co.','Splaiul Crizantemelor 3, Bucureşti, 500100, România','0733999000');

ID_FURNIZOR	NUMEFURNIZOR	ADRESAFURNIZOR	TELEFONFURNIZOR
1	44 Fresco Verde	Strada Trandafirilor 10, Bucureşti, 010101, România	0721123456
2	45 Happy Flower Holland	Aleea Florilor 5, Bucuresti, 400000, România	0765987654
3	46 GreenLeaf Plants	Bulevardul Liliacului 25, Bucureşti, 700000, România	0744555888
4	47 Natures Garden Supplies	Strada Irisului 8, Bucureşti, 300001, România	0782111222
5	48 Bloom Nursery Co.	Splaiul Crizantemelor 3, Bucureşti, 500100, România	0733999000

Codul:

```
create table FURNIZOR
(id_furnizor number(5) constraint pkey_furnizor primary key,
NumeFurnizor varchar(30) constraint NumeFurnizor not null,
AdresaFurnizor varchar(100) constraint AdresaFurnizor not null unique,
TelefonFurnizor varchar(10) constraint TelefonFurnizor not null unique
);

insert into FURNIZOR
values (flor_seq.nextval,'Fresco Verde','Strada Trandafirilor 10, Bucureşti, 010101, România','0721123456');

insert into FURNIZOR
values (flor_seq.nextval,'Happy Flower Holland','Aleea Florilor 5, Bucureşti, 400000, România','0765987654');

insert into FURNIZOR
values (flor_seq.nextval,'GreenLeaf Plants','Bulevardul Liliacului 25, Bucureşti, 700000, România','0744555888');

insert into FURNIZOR
values (flor_seq.nextval,'Natures Garden Supplies','Strada Irisului 8, Bucureşti, 300001, România','0782111222');

insert into FURNIZOR
values (flor_seq.nextval,'Bloom Nursery Co.','Splaiul Crizantemelor 3, Bucureşti, 500100, România','0733999000');
```

- TABELA PRODUS:

create table PRODUS

```
(id_produs number(5) constraint pkey_produs primary key,
 id_categprod number(5), constraint fk_produs_categprod foreign
 key(id_categprod) references CATEGORIE_PRODUS(id_categprod),
 id_furnizor number(5), constraint fk_produs_furnizor foreign
 key(id_furnizor) references FURNIZOR(id_furnizor),
 NumeProdus varchar(30) constraint NumeProdus not null,
 Pret number(5) constraint Pret not null
);
```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_PRODUS	NUMBER (5, 0)	No	(null)	1 (null)	
2 ID_CATEGPROD	NUMBER (5, 0)	Yes	(null)	2 (null)	
3 ID_FURNIZOR	NUMBER (5, 0)	Yes	(null)	3 (null)	
4 NUMEPRODUS	VARCHAR2 (30 BYTE)	No	(null)	4 (null)	
5 PRET	NUMBER (5, 0)	No	(null)	5 (null)	

insert into PRODUS

```
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS
where NumeCategorie='Buchete'), (select id_furnizor from FURNIZOR where
NumeFurnizor='Fresco Verde'), 'Buchet trandafiri', 100);
```

insert into PRODUS

```
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS
where NumeCategorie='Buchete'), (select id_furnizor from FURNIZOR where
NumeFurnizor='Fresco Verde'), 'Buchet de vara', 80);
```

```
insert into PRODUS  
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS  
where NumeCategorie='Buchete'), (select id_furnizor from FURNIZOR where  
NumeFurnizor='Fresco Verde'), 'Buchet de primavara', 120);
```

```
insert into PRODUS  
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS  
where NumeCategorie='Plante de interior'), (select id_furnizor from  
FURNIZOR where NumeFurnizor='Natures Garden Supplies'), 'Dracena', 50);
```

```
insert into PRODUS  
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS  
where NumeCategorie='Plante de interior'), (select id_furnizor from  
FURNIZOR where NumeFurnizor='Happy Flower Holland'), 'Ficus lyrata', 30);
```

```
insert into PRODUS  
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS  
where NumeCategorie='Plante de interior'), (select id_furnizor from  
FURNIZOR where NumeFurnizor='GreenLeaf Plants'), 'Trandafir japonez', 50);
```

```
insert into PRODUS  
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS  
where NumeCategorie='Plante de exterior'), (select id_furnizor from  
FURNIZOR where NumeFurnizor='Bloom Nursery Co.'), 'Trandafir englezesc',  
60);
```

insert into PRODUS
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS
where NumeCategorie='Plante de exterior'), (select id_furnizor from
FURNIZOR where NumeFurnizor='GreenLeaf Plants'), 'Cupressus
semperfirens', 80);

insert into PRODUS
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS
where NumeCategorie='Plante de exterior'), (select id_furnizor from
FURNIZOR where NumeFurnizor='Natures Garden Supplies'), 'Bujor', 24);

insert into PRODUS
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS
where NumeCategorie='Accesorii'), (select id_furnizor from FURNIZOR where
NumeFurnizor='Happy Flower Holland'), 'Ghiveci ceramic', 25);

insert into PRODUS
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS
where NumeCategorie='Accesorii'), (select id_furnizor from FURNIZOR where
NumeFurnizor='Natures Garden Supplies'), 'Vaza de sticla', 25);

insert into PRODUS
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS
where NumeCategorie='Accesorii'), (select id_furnizor from FURNIZOR where
NumeFurnizor='Bloom Nursery Co.'), 'Felicitare', 15);

insert into PRODUS

```
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS  
where NumeCategorie='La bucata'), (select id_furnizor from FURNIZOR where  
NumeFurnizor='GreenLeaf Plants'), 'Trandafir roz', 18);
```

insert into PRODUS

```
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS  
where NumeCategorie='La bucata'), (select id_furnizor from FURNIZOR where  
NumeFurnizor='Natures Garden Supplies'), 'Crin alb', 14);
```

insert into PRODUS

```
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS  
where NumeCategorie='La bucata'), (select id_furnizor from FURNIZOR where  
NumeFurnizor='Happy Flower Holland'), 'Frezie galbenă', 10);
```

ID_PRODUS	ID_CATEGPROD	ID_FURNIZOR	NUMEPRODUS	PRET
1	49	32	44 Buchet trandafiri	100
2	50	32	44 Buchet de vara	80
3	51	32	44 Buchet de primavara	120
4	52	33	47 Dracena	50
5	53	33	45 Ficus lyrata	30
6	54	33	46 Trandafir japonez	50
7	55	34	48 Trandafir englezesc	60
8	56	34	46 Cupressus sempervirens	80
9	57	34	47 Bujor	24
10	58	35	45 Ghiveci ceramic	25
11	59	35	47 Vaza de sticla	25
12	60	35	48 Felicitare	15
13	61	36	46 Trandafir roz	18
14	62	36	47 Crin alb	14
15	63	36	45 Frezie galbenă	10

Codul:

```
Worksheet Query Builder
create table PRODUS
(
    id_produs number(5) constraint pkey_produs primary key,
    id_categprod number(5), constraint fk_produs_categprod foreign key(id_categprod) references CATEGORIE_PRODUS(id_categprod),
    id_furnizor number(5), constraint fk_produs_furnizor foreign key(id_furnizor) references FURNIZOR(id_furnizor),
    NumeProdus varchar(30) constraint NumeProdus not null,
    Pret number(5) constraint Pret not null
);

insert into PRODUS
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS where NumeCategoria='Buchete'), (select id_furnizor from FURNIZOR where NumeFurnizor='Fresco Verde'), 'Buchet trandafiri', 100);

insert into PRODUS
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS where NumeCategoria='Buchete'), (select id_furnizor from FURNIZOR where NumeFurnizor='Fresco Verde'), 'Buchet de vara', 80);

insert into PRODUS
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS where NumeCategoria='Buchete'), (select id_furnizor from FURNIZOR where NumeFurnizor='Fresco Verde'), 'Buchet de primavara', 120);

insert into PRODUS
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS where NumeCategoria='Plante de interior'), (select id_furnizor from FURNIZOR where NumeFurnizor='Natures Garden Supplies'), 'Dracena', 50);

insert into PRODUS
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS where NumeCategoria='Plante de interior'), (select id_furnizor from FURNIZOR where NumeFurnizor='Happy Flower Holland'), 'Ficus lyrata');

insert into PRODUS
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS where NumeCategoria='Plante de interior'), (select id_furnizor from FURNIZOR where NumeFurnizor='GreenLeaf Plants'), 'Trandafir japonez');

insert into PRODUS
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS where NumeCategoria='Plante de exterior'), (select id_furnizor from FURNIZOR where NumeFurnizor='Bloom Nursery Co.'), 'Trandafir englezesc');

insert into PRODUS
values (flor_seq.nextval, (select id_categprod from CATEGORIE_PRODUS where NumeCategoria='Plante de exterior'), (select id_furnizor from FURNIZOR where NumeFurnizor='GreenLeaf Plants'), 'Cupressus sempervir');
```

• TABELA DETALII_COMANDA:

create table DETALII_COMANDA

(id_detaliicomanda number(5) constraint pkey_detaliicomanda primary key,
id_comanda number(5), constraint fk_detaliicomanda_comanda foreign
key(id_comanda) references COMANDA(id_comanda),
id_produs number(5), constraint fk_detaliicomanda_produs foreign
key(id_produs) references PRODUS(id_produs),

NumarProduse number(5) constraint NumarProduse not null

);

The screenshot shows the Oracle SQL Developer interface with the 'detaliicomanda' table selected. The table has four columns: ID_DETALIIICOMANDA, ID_COMANDA, ID_PRODUS, and NUMARPRODUSE. The first three columns are of type NUMBER(5,0) and the last one is NUMBER(5,0). The table has four rows.

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_DETALIIICOMANDA	NUMBER(5,0)	No	(null)	1	(null)
2 ID_COMANDA	NUMBER(5,0)	Yes	(null)	2	(null)
3 ID_PRODUS	NUMBER(5,0)	Yes	(null)	3	(null)
4 NUMARPRODUSE	NUMBER(5,0)	No	(null)	4	(null)

```
insert into DETALII_COMANDA  
values (flor_seq.nextval,27,(select id_produs from PRODUS where  
NumeProdus='Cupressus sempervirens'),1);
```

```
insert into DETALII_COMANDA  
values (flor_seq.nextval,27,(select id_produs from PRODUS where  
NumeProdus='Ghiveci ceramic'),1);
```

```
insert into DETALII_COMANDA  
values (flor_seq.nextval,28,(select id_produs from PRODUS where  
NumeProdus='Buchet trandafiri'),1);
```

```
insert into DETALII_COMANDA  
values (flor_seq.nextval,29,(select id_produs from PRODUS where  
NumeProdus='Frezie galbenă'),15);
```

```
insert into DETALII_COMANDA  
values (flor_seq.nextval,29,(select id_produs from PRODUS where  
NumeProdus='Vaza de sticla'),1);
```

```
insert into DETALII_COMANDA  
values (flor_seq.nextval,29,(select id_produs from PRODUS where  
NumeProdus='Felicitare'),1);
```

```
insert into DETALII_COMANDA  
values (flor_seq.nextval,30,(select id_produs from PRODUS where  
NumeProdus='Ghiveci ceramic'),3);
```

```
insert into DETALII_COMANDA  
values (flor_seq.nextval,30,(select id_produs from PRODUS where  
NumeProdus='Trandafir englezesc'),1);
```

```
insert into DETALII_COMANDA  
values (flor_seq.nextval,31,(select id_produs from PRODUS where  
NumeProdus='Dracena'),1);
```

```
insert into DETALII_COMANDA  
values (flor_seq.nextval,31,(select id_produs from PRODUS where  
NumeProdus='Bujor'),2);
```

ID_DETALIICOMANDA	ID_COMANDA	ID_PRODUS	NUMARPRODUSE
1	64	27	56
2	65	27	58
3	66	28	49
4	67	29	63
5	68	29	59
6	69	29	60
7	70	30	58
8	71	30	55
9	72	31	52
10	73	31	57

Codul:

```
create table DETALII_COMANDA
(id_detalii_comanda number(5) constraint pkey_detalii_comanda primary key,
id_comanda number(5), constraint fk_detalii_comanda_comanda foreign key(id_comanda) references COMANDA(id_comanda),
id_produs number(5), constraint fk_detalii_comanda_produs foreign key(id_produs) references PRODUS(id_produs),
NumarProduse number(5) constraint NumarProduse not null
);

insert into DETALII_COMANDA
values (flor_seq.nextval,27,(select id_produs from PRODUS where NumeProdus='Cupressus sempervirens'),1);

insert into DETALII_COMANDA
values (flor_seq.nextval,27,(select id_produs from PRODUS where NumeProdus='Ghiveci ceramic'),1);

insert into DETALII_COMANDA
values (flor_seq.nextval,28,(select id_produs from PRODUS where NumeProdus='Buchet trandafiri'),1);

insert into DETALII_COMANDA
values (flor_seq.nextval,29,(select id_produs from PRODUS where NumeProdus='Frezie galbenă'),15);

insert into DETALII_COMANDA
values (flor_seq.nextval,29,(select id_produs from PRODUS where NumeProdus='Vaza de sticla'),1);

insert into DETALII_COMANDA
values (flor_seq.nextval,29,(select id_produs from PRODUS where NumeProdus='Felicitare'),1);

insert into DETALII_COMANDA
values (flor_seq.nextval,30,(select id_produs from PRODUS where NumeProdus='Ghiveci ceramic'),3);

insert into DETALII_COMANDA
values (flor_seq.nextval,30,(select id_produs from PRODUS where NumeProdus='Trandafir englezesc'),1);

insert into DETALII_COMANDA
```

Cerinta 12:

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

- subcereri sincronizate în care intervin cel puțin 3 tabele
- subcereri nesincronizate în clauza FROM
- grupări de date cu subcereri nesincronizate în care intervin cel puțin 3 tabele, funcții grup, filtrare la nivel de grupuri (în cadrul aceleiasi cereri)
- ordonări și utilizarea funcțiilor NVL și DECODE (în cadrul aceleiasi cereri)
- utilizarea a cel puțin 2 funcții pe siruri de caractere, 2 funcții pe date calendaristice, a cel puțin unei expresii CASE
- utilizarea a cel puțin 1 bloc de cerere (clauza WITH)

Observație: Într-o cerere se vor regăsi mai multe elemente dintre cele enumerate mai sus, astfel încât cele 5 cereri să le cuprindă pe toate.

Exercitiul 1:

Intr-o companie se dorește generarea unui raport cu informații relevante despre angajați. Pentru fiecare angajat, se dorește afisarea numelui lui întreg cu majuscule, dacă numele este scurt sau lung (un nume lung este considerat a avea mai mult de 16 caractere), anul angajării, cate luni complete a lucrat la firma, salariul și urmatoarea zi de salariu. Pentru angajații cu un salariu mai mare de 6000, data următoarei plăti va fi ultima zi a lunii curente, în timp ce pentru ceilalți angajați va fi prima zi de luni după ultima zi a lunii curente. Generați acest răspuns folosind și o clauza with.

Prin acest exercițiu răspundem la cerințele:

- utilizarea a cel puțin 2 funcții pe siruri de caractere, 2 funcții pe date calendaristice, a cel puțin unei expresii CASE
- utilizarea a cel puțin 1 bloc de cerere (clauza WITH)

with InformatiiCerute as(

```
select rpad(upper(NumeAngajat || ' ' || PrenumeAngajat),25) "Angajat",
      case
        when length(NumeAngajat || ' ' || PrenumeAngajat)>16 then 'Nume lung'
        else 'Nume scurt'
      end as "Lungimea numelui",
      extract(year from DataAngajare) "An angajare",
      round(months_between(sysdate,DataAngajare)) "Luni de munca la firma",
      Salariu "Salariu",
      case
        when salariu>6000 then last_day(sysdate)
        else next_day(last_day(sysdate),'Monday')
      end as "Urmatoarea zi de salariu"
    from ANGAJAT,dual
)
select *
from InformatiiCerute;
```

```

with InformatiiCerute as(
    select rpad(upper(NumeAngajat || ' ' || PrenumeAngajat),25) "Angajat",
    case
        when length(NumeAngajat || ' ' || PrenumeAngajat)>16 then 'Nume lung'
        else 'Nume scurt'
    end as "Lungimea numelui",
    extract(year from DataAngajare) "An angajare",
    round(months_between(sysdate,DataAngajare)) "Luni de munca la firma",
    Salariu "Salariu",
    case
        when salariu>6000 then last_day(sysdate)
        else next_day(last_day(sysdate),'Monday')
    end as "Urmatoarea zi de salariu"
    from ANGAJAT,dual
)
select *
from InformatiiCerute;

```

Angajat	Lungimea numelui	An angajare	Luni de munca la firma	Salariu	Urmatoarea zi de salariu
1 IONESCU ANASTASIA	Nume lung	2019	45	5000	05-JUN-23
2 FLORESCU LETITIA	Nume scurt	2019	47	5100	05-JUN-23
3 STANESCU STEFAN	Nume scurt	2005	210	9000	31-MAY-23
4 DUMITRACHE RAFAEL	Nume lung	2019	51	6200	31-MAY-23
5 TEODORESCU ELEONORA	Nume lung	2015	97	6400	31-MAY-23

Exercitiul 2:

Se doreste sa se afiseze numele orasului, numele strazii, numarul locuintei, informatii despre bloc (in cazul in care exista, iar daca nu exista sa se afiseze 'Nespecificat'), informatii despre scara (in cazul in care exista, iar daca nu exista sa se afiseze 'Nespecificat'), informatii despre etaj (in cazul in care exista, iar daca nu exista sa se afiseze 'Nespecificat'), informatii despre apartament (in cazul in care exista, iar daca nu exista sa se afiseze 'Nespecificat') si codul postal al fiecarei adrese din baza de date. Datele sa fie ordonate ascendent in functie de coloana „Strada”.

Prin acest exercitiu raspundem la cerinta:

-ordonări si utilizarea functiilor NVL și DECODE (in cadrul aceleiasi cereri)

```

select rpad(oras,10) "Oras" ,strada "Strada" ,numar "Numar",nvl(bloc,'Nespecificat')
"Bloc",
    nvl(to_char(scara),'Nespecificat') "Scara",decode(etaj, NULL, 'Nespecificat', etaj) "Etaj",
    decode(apartament, NULL, 'Nespecificat', apartament) "Apartament",codpostal "Cod
postal"
from adresa
order by strada asc;

select rpad(oras,10) "Oras" ,strada "Strada" ,numar "Numar",nvl(bloc,'Nespecificat') "Bloc",
    nvl(to_char(scara),'Nespecificat') "Scara",decode(etaj, NULL, 'Nespecificat', etaj) "Etaj",
    decode(apartament, NULL, 'Nespecificat', apartament) "Apartament",codpostal "Cod postal"
from adresa
order by strada asc;

```

Oras	Strada	Numar	Bloc	Scara	Etaj	Apartament	Cod postal
1 Constanta	Bd. Mamaia	20	Nespecificat	Nespecificat	Nespecificat	Nespecificat	900001
2 Cluj-Napoc	Str. Bucuriei	15 A	2	3	7		400000
3 Timisoara	Str. Mihai Viteazu	8 B3	1	6	60		300100
4 Iasi	Str. Stefan Cel Mare	25	Nespecificat	Nespecificat	Nespecificat	Nespecificat	700300
5 Bucuresti	Str. Victoriei	10	Nespecificat	Nespecificat	Nespecificat	Nespecificat	010101

Exercitiul 3:

Se doreste obtinerea unei raportari a totalului de produse achizitionate de catre clienti in categoria „Accesorii”. Pentru fiecare client se vor afisa numele clientului, numele produsului si totalul de produse achizitionate. Raportul trebuie sa includa doar clientii ai caror totaluri de produse achizitionate sunt mai mici decat media generala a numarului de produse achizitionate.

Prin acest exercitiu raspundem la cerinta:

- grupări de date cu subcereri nesincronizate in care intervin cel putin 3 tabele, functii grup, filtrare la nivel de grupuri (in cadrul aceleiasi cereri)

```

select c.NumeClient, p.NumeProdus, sum(dc.NumarProduse) "Total Produse"
from client c
join comanda co on c.id_client = co.id_client
join detalii_comanda dc on co.id_comanda = dc.id_comanda
join produs p on dc.id_produs = p.id_produs
where p.id_categprod in (
    select id_categprod
    from categorie_produs
    where NumeCategorie = 'Accesorii'
)
group by c.NumeClient, p.NumeProdus
having sum(dc.NumarProduse) < (
    select avg(NumarProduse)
    from detalii_comanda
);

```

```

select c.NumeClient, p.NumeProdus, sum(dc.NumarProduse) "Total Produse"
from client c
join comanda co on c.id_client = co.id_client
join detalii_comanda dc on co.id_comanda = dc.id_comanda
join produs p on dc.id_produs = p.id_produs
where p.id_categprod in (
    select id_categprod
    from categorie_produs
    where NumeCategorie = 'Accesorii'
)
group by c.NumeClient, p.NumeProdus
having sum(dc.NumarProduse) < (
    select avg(NumarProduse)
    from detalii_comanda
);

```

NUMECLIENT	NUMEPERODUS	Total Produse
1 Nica	Felicitare	1
2 Feurdean	Ghiveci ceramic	1
3 Nica	Vaza de sticla	1

Exercitiul 4:

Dorim sa obtinem numele produselor, categoria produsului si numele furnizorului pentru produsele furnizate de furnizorul al carui nume contine secevanta „Natures Garden”.

Prin acest exercitiu raspundem la cerinta:

- subcereri nesincronizate în clauza FROM

```
select p.NumeProdus, c.NumeCategorie, f.NumeFurnizor
```

```
from (
```

```
    select id_produs, id_categprod, id_furnizor, numeprodus
```

```
        from Produs
```

```
) p,
```

```
(
```

```
    select id_categprod, NumeCategorie
```

```
        from Categorie_Produs
```

```
) c,
```

```
(
```

```
    select id_furnizor, NumeFurnizor
```

```
        from Furnizor
```

```
) f
```

```
where p.id_categprod = c.id_categprod
```

```
and p.id_furnizor = f.id_furnizor
```

```
and f.numefurnizor like '%Natures Garden%';
```

```

select p.NumeProdus, c.NumeCategorie, f.NumeFurnizor
from (
    select id_produs, id_categprod, id_furnizor, numeprodus
    from Produs
) p,
(
    select id_categprod, NumeCategorie
    from Categorie_Produs
) c,
(
    select id_furnizor, NumeFurnizor
    from Furnizor
) f
where p.id_categprod = c.id_categprod
and p.id_furnizor = f.id_furnizor
and f.numefurnizor like '%Natures Garden%';

```

NUMEPRODUS	NUMECATEGORIE	NUMEFURNIZOR
1 Dracena	Plante de interior	Natures Garden Supplies
2 Bujor	Plante de exterior	Natures Garden Supplies
3 Vaza de sticla	Accesorii	Natures Garden Supplies
4 Crin alb	La bucata	Natures Garden Supplies

Exercitiul 5:

Obtineti numarul de recenzii, numarul total de produse cumparate, precum si numarul de produse cumparate de la furnizorul 'Natures Garden' pentru fiecare client. Afisati rezultatele alaturi de numele si prenumele fiecarui client.

Prin acest exercitiu raspundem la cerinta:

- subcereri sincronizate in care intervin cel putin 3 tabele

```
select c.NumeClient || ' ' || c.PrenumeClient as "Client",
```

```
(select count(*)
```

```
from recenzie r
```

```
where r.id_client = c.id_client) as "NumarRecenzi",
```

```
(select count(dc.id_produs)
```

```
from detalii_comanda dc
```

```

inner join comanda co on co.id_comanda = dc.id_comanda
inner join produs p on p.id_produs = dc.id_produs
inner join furnizor f on f.id_furnizor = p.id_furnizor
where co.id_client = c.id_client) as "Numar produse cumpарате",
(select count(dc.id_produs)
from detalii_comanda dc
inner join comanda co on co.id_comanda = dc.id_comanda
inner join produs p on p.id_produs = dc.id_produs
inner join furnizor f on f.id_furnizor = p.id_furnizor
where f.NumeFurnizor like '%Natures%'
and co.id_client = c.id_client) as "Numar produse cumpарате de la Natures Garden"
from client c;

```

```

select c.NumeClient || ' ' || c.PrenumeClient as "Client",
(select count(*)
from recenzie r
where r.id_client = c.id_client) as "NumarRecenzii",
(select count(dc.id_produs)
from detalii_comanda dc
inner join comanda co on co.id_comanda = dc.id_comanda
inner join produs p on p.id_produs = dc.id_produs
inner join furnizor f on f.id_furnizor = p.id_furnizor
where co.id_client = c.id_client) as "Numar produse cumpарате",
(select count(dc.id_produs)
from detalii_comanda dc
inner join comanda co on co.id_comanda = dc.id_comanda
inner join produs p on p.id_produs = dc.id_produs
inner join furnizor f on f.id_furnizor = p.id_furnizor
where f.NumeFurnizor like '%Natures%'
and co.id_client = c.id_client) as "Numar produse cumpарате de la Natures Garden"
from client c;

```

Client	NumarRecenzi	Numar produse cumparate	Numar produse cumparate de la Natures Garden
1 Popescu Ioana	1	2	0
2 Apostol Teodora	0	0	0
3 Olteanu Stefan	0	0	0
4 Tintea Alexandru	1	2	2
5 Nica Ana	1	4	1
6 Feurdean Mihai	2	2	0

Cerinta 13:

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

1) UPDATE

Se mareste salariul angajatilor cu rolul de „Florar”.

```

update angajat
set salariu = salariu * 1.1
where id_angajat in (
    select id_angajat
    from angajat
    where id_rol=(select id_rol
                  from rol_angajat
                  where numerol = 'Florar')
);

```

The screenshot shows the MySQL Workbench interface with a script editor containing the provided SQL code. Below the editor, the 'Query Result' tab shows the message 'Task completed in 0.026 seconds'. At the bottom, it indicates '2 rows updated.'

update angajat

set salariu = salariu * 1.1

where id_angajat in (

select id_angajat

from angajat

where id_rol=(select id_rol

from rol_angajat

where numerol = 'Florar'));

ID_ANGAJAT	ID_ROL	NUMEANGAJAT	PRENUMEANGAJAT	DATAANGAJARE	SALARIU
1	6	Ionescu	Anastasia	16-AUG-19	5000
2	7	Florescu	Letitia	04-JUL-19	5100
3	8	Stanescu	Stefan	01-DEC-05	9000
4	9	Dumitache	Rafael	10-MAR-19	6200
5	10	Teodorescu	Eleonora	15-APR-15	6400



ID_ANGAJAT	ID_ROL	NUMEANGAJAT	PRENUMEANGAJAT	DATAANGAJARE	SALARIU
1	6	Ionescu	Anastasia	16-AUG-19	5500
2	7	Florescu	Letitia	04-JUL-19	5610
3	8	Stanescu	Stefan	01-DEC-05	9000
4	9	Dumitache	Rafael	10-MAR-19	6200
5	10	Teodorescu	Eleonora	15-APR-15	6400

2) UPDATE

Se micsoreaza pretul fiecarui produs din categoria „Accesorii”.

```

update produs
set pret = pret * 0.95
where id_produs in (
    select id_produs
    from produs
    where id_categprod=(select id_categprod
                            from categorie_produs
                            where numecategorie = 'Accesorii')
);

```

Script Output x | Query Result x
✖ ✔ ✖ ✖ Task completed in 0.024 seconds

3 rows updated.

update produs

set pret = pret * 0.95

where id_produs in (

```

select id_produs
from produs
where id_categprod=(select id_categprod
from categorie_produs
where numecategorie = 'Accesorii')
);

```

	ID_PRODUS	ID_CATEGPROD	ID_FURNIZOR	NUMEPRODUS	PRET
1	49	32	44	Buchet trandafiri	100
2	50	32	44	Buchet de vara	80
3	51	32	44	Buchet de primavara	120
4	52	33	47	Dracena	50
5	53	33	45	Ficus lyrata	30
6	54	33	46	Trandafir japonez	50
7	55	34	48	Trandafir englezesc	60
8	56	34	46	Cupressus sempervirens	80
9	57	34	47	Bujor	24
10	58	35	45	Ghiveci ceramic	25
11	59	35	47	Vaza de sticla	25
12	60	35	48	Felicitare	15
13	61	36	46	Trandafir roz	18
14	62	36	47	Crin alb	14
15	63	36	45	Frezie galbenă	10



	ID_PRODUS	ID_CATEGPROD	ID_FURNIZOR	NUMEPRODUS	PRET
1	49	32	44	Buchet trandafiri	100
2	50	32	44	Buchet de vara	80
3	51	32	44	Buchet de primavara	120
4	52	33	47	Dracena	50
5	53	33	45	Ficus lyrata	30
6	54	33	46	Trandafir japonez	50
7	55	34	48	Trandafir englezesc	60
8	56	34	46	Cupressus sempervirens	80
9	57	34	47	Bujor	24
10	58	35	45	Ghiveci ceramic	24
11	59	35	47	Vaza de sticla	24
12	60	35	48	Felicitare	14
13	61	36	46	Trandafir roz	18
14	62	36	47	Crin alb	14
15	63	36	45	Frezie galbenă	10

3) DELETE

Se sterg clientii care nu au plasat comenzi.

```
delete client c
where not exists (
    select *
    from comanda cmd
    where cmd.id_client = c.id_client );
```

Script Output Query Result Task completed in 0.042 seconds

Action:

2 rows deleted.

delete client c

where not exists (

select *

from comanda cmd

where cmd.id_client = c.id_client);

ID_CLIENT	ID_ADRESA	NUMECLIENT	PRENUMECLIENT	EMAILCLIENT	TELEFONCLIENT
1	16	11 Popescu	Ioana	ioanapopescu@gmail.com	0744322188
2	17	12 Apostol	Teodora	teodoraapostol@yahoo.com	0744325438
3	18	13 Olteanu	Stefan	stefanolteanu@gmail.com	0734522188
4	19	14 Tintea	Alexandru	alextintea@gmail.com	0744678388
5	20	15 Nica	Ana	ananica@ymail.com	0722722722
6	21	11 Feurdean	Mihai	mihaifeur@gmail.com	0747777788



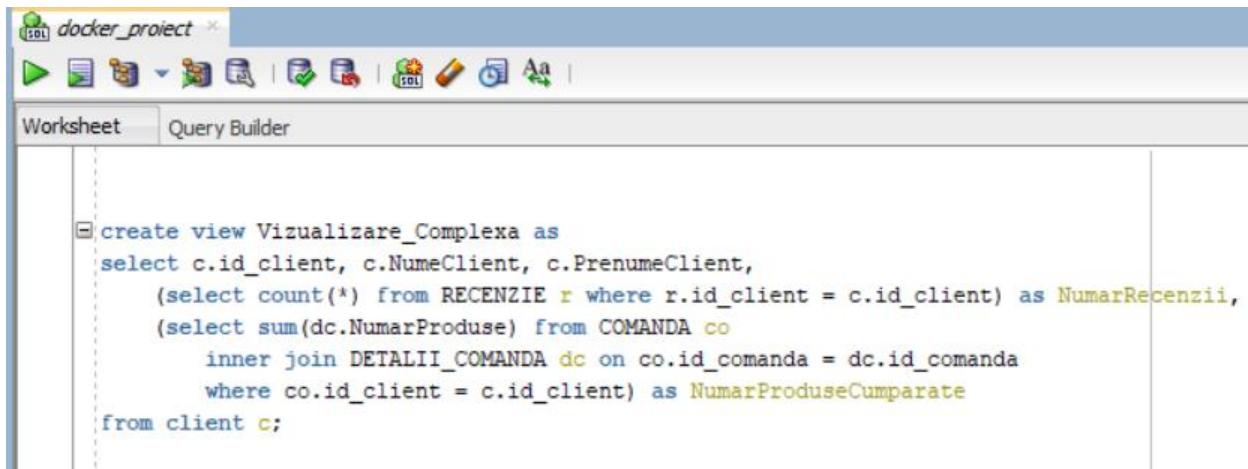
ID_CLIENT	ID_ADRESA	NUMECLIENT	PRENUMECLIENT	EMAILCLIENT	TELEFONCLIENT
1	16	11 Popescu	Ioana	ioanapopescu@gmail.com	0744322188
2	19	14 Tintea	Alexandru	alextintea@gmail.com	0744678388
3	20	15 Nica	Ana	ananica@ymail.com	0722722722
4	21	11 Feurdean	Mihai	mihaifeur@gmail.com	0747777788

Cerinta 14:

Crearea unei vizualizări complexe. Dați un exemplu de operație LMD permisă pe vizualizarea respectivă și un exemplu de operație LMD nepermisă.

Un exemplu de vizualizare complexă este urmatorul:

```
create view Vizualizare_Complexa as
select c.id_client, c.NumeClient, c.PrenumeClient,
       (select count(*) from RECENZIE r where r.id_client = c.id_client) as NumarRecenzii,
       (select sum(dc.NumarProduse) from COMANDA co
        inner join DETALII_COMANDA dc on co.id_comanda = dc.id_comanda
        where co.id_client = c.id_client) as NumarProduseCumparate
from client c;
```



The screenshot shows a database management system window titled "docker_project". The toolbar contains various icons for file operations, search, and help. Below the toolbar, there are two tabs: "Worksheet" (selected) and "Query Builder". The main area displays the SQL code for creating a view named "Vizualizare_Complexa". The code uses subqueries to calculate the number of reviews and the total number of items purchased by each client. The code is as follows:

```
create view Vizualizare_Complexa as
select c.id_client, c.NumeClient, c.PrenumeClient,
       (select count(*) from RECENZIE r where r.id_client = c.id_client) as NumarRecenzii,
       (select sum(dc.NumarProduse) from COMANDA co
        inner join DETALII_COMANDA dc on co.id_comanda = dc.id_comanda
        where co.id_client = c.id_client) as NumarProduseCumparate
from client c;
```

View VIZUALIZARE_COMPLEXA created.

Aceasta vizualizare complexa contine ID-ul, numele si prenumele clientilor, numarul de recenzii de la fiecare client si numarul total de produse cumparate de fiecare client.

Un exemplu de operatie LMD permisa pe vizualizarea anterioara, „Vizualizare_Complexa”, este:

```
select *
from Vizualizare_Complexa
where NumarRecenzii >= 2;
```

```
select *
from Vizualizare_Complexa
where NumarRecenzii >= 2;
```

The screenshot shows a database query results window. At the top, there are tabs for 'Script Output' and 'Query Result'. Below the tabs, there are icons for refresh, print, and other database operations. The status bar indicates 'All Rows Fetched: 1 in 0.007 seconds'. The main area displays a table with four columns: ID_CLIENT, NUMECLIENT, PRENUMECCLIENT, and NUMARRECENZII. The single row of data is: 1, 21 Feurdean, Mihai, 2. There is also a column labeled 'NUMARPRODUSECUMPARATE' which has a value of 2.

ID_CLIENT	NUMECLIENT	PRENUMECCLIENT	NUMARRECENZII	NUMARPRODUSECUMPARATE
1	21 Feurdean	Mihai	2	2

Aceasta interogare permite selectarea tuturor inregistrarilor din vizualizarea „Vizualizare_Complexa”, afisand informatiile despre id_client, NumeClient, PrenumeClient, NumarRecenzii si NumarProduseCumparate pentru fiecare client care a trimis un numar mai mare sau egal cu 2 de recenzii.

Un exemplu de operatie LMD nepermisa pe vizualizarea anterioara, „Vizualizare_Complexa”, este:

```
insert into Vizualizare_Complexa (id_client, NumeClient, PrenumeClient,
NumarRecenzii, NumarProduseCumparate)
values (500, 'Jane', 'Doe', 3, 10);
```

```
insert into Vizualizare_Complexa (id_client, NumeClient, PrenumeClient, NumarRecenzii, NumarProduseCumparate)
values (500, 'Jane', 'Doe', 3, 10);
```

Aceasta interogare incearca sa insereze o noua inregistrare in vizualizarea complexa „Vizualizare_Complexa”. Totusi, deoarece vizualizarile nu au o structura fizica si sunt rezultatul unei interogari, nu este permisa inserarea directa de date intr-o vizualizare. In schimb, trebuie sa se insereze insertii in tabelele de baza corespunzatoare pentru a actualiza vizualizarea in mod corespunzator.

```
Error starting at line : 19 in command -
insert into Vizualizare_Complexa (id_client, NumeClient, PrenumeClient, NumarRecenzii, NumarProduseCumparate)
values (500, 'Jane', 'Doe', 3, 10)
Error at Command Line : 19 Column : 73
Error report -
SQL Error: ORA-01733: virtual column not allowed here
01733. 00000 - "virtual column not allowed here"
*Cause:
*Action:
```

Cerinta 15:

Formulați în limbaj natural și implementați în SQL: o cerere ce utilizează operația outer-join pe minimum 4 tabele, o cerere ce utilizează operația division și o cerere care implementează analiza top-n.

Exercitiul 1 (left outer join):

Afiseaza toate detaliile despre clienti, comenzile lor, detaliile de comanda, produsele si categoriile de produse asociate acestora.

```
select *
from client cl
left join comanda co on cl.id_client = co.id_client
left join detalii_comanda dc on dc.id_comanda=co.id_comanda
left join produs p on dc.id_produs = p.id_produs
left join categorie_produs cp on cp.id_categprod=p.id_categprod;
```

```

select *
from client cl
left join comanda co on cl.id_client = co.id_client
left join detalii_comanda dc on dc.id_comanda=co.id_comanda
left join produs p on dc.id_produs = p.id_produs
left join categorie_produs cp on cp.id_categprod=p.id_categprod;

```

	ID_CLIENT	ID_ADRESA	NUMECLIENT	PRENUMECIENT	EMAILCLIENT	TELEFONCLIENT	ID_COMANDA	ID_ANGAJAT	ID_CLIENT_1	DATACOMANDA	TIPLATA
1	20	15 Nica	Ana	Ana	ananica@ymail.com	0722722722		28	6	20 05-MAY-20	card
2	19	14 Tintea	Alexandru	Alexandru	alextintea@gmail.com	0744678388		31	9	19 04-JUN-21	cash
3	16	11 Popescu	Ioana	Ioana	ioanapopescu@gmail.com	0744322188		30	10	16 04-JUN-20	cash
4	21	11 Feurdean	Mihai	Mihai	mihafeur@gmail.com	0747777788		27	6	21 15-DEC-20	card
5	19	14 Tintea	Alexandru	Alexandru	alextintea@gmail.com	0744678388		31	9	19 04-JUN-21	cash
5	21	11 Feurdean	Mihai	Mihai	mihafeur@gmail.com	0747777788		27	6	21 15-DEC-20	card
7	16	11 Popescu	Ioana	Ioana	ioanapopescu@gmail.com	0744322188		30	10	16 04-JUN-20	cash
8	20	15 Nica	Ana	Ana	ananica@ymail.com	0722722722		29	7	20 27-NOV-20	cash

TIPLATA	ID_DETALIICOMANDA	ID_COMANDA_1	ID_PRODUS	NUMARPRODUSE	ID_PRODUS_1	ID_CATEGPROD	ID_FURNIZOR	NUMEPRODUS
card	66	28	49	1	49	32	44	Buchet trandafiri
cash	72	31	52	1	52	33	47	Dracena
cash	71	30	55	1	55	34	48	Trandafir englezesc
card	64	27	56	1	56	34	46	Cupressus sempervirens
cash	73	31	57	2	57	34	47	Bujor
card	65	27	58	1	58	35	45	Ghiveci ceramic
cash	70	30	58	3	58	35	45	Ghiveci ceramic
cash	68	29	59	1	59	35	47	Vaza de sticla

PRET	ID_CATEGPROD_1	NUMECATEGORIE	DESCRIERE
100	32	Buchete	Buchete de flori proaspete și aranjamente florale elegante.
50	33	Plante de interior (null)	
60	34	Plante de exterior	Plante de exterior perene și anuale
80	34	Plante de exterior	Plante de exterior perene și anuale
24	34	Plante de exterior	Plante de exterior perene și anuale
24	35	Accesorii	Ghivece, ambalaje, felicitari, ingrasaminte, decoratiuni etc.
24	35	Accesorii	Ghivece, ambalaje, felicitari, ingrasaminte, decoratiuni etc.
24	35	Accesorii	Ghivece, ambalaje, felicitari, ingrasaminte, decoratiuni etc.

Exercitiul 2 (division):

Afiseaza ID-ul, numele si prenumele angajatilor care nu au toate rolurile atribuite (trebuie sa fie toti angajatii pentru ca fiecare are numai cate un rol, deci niciodata pe toate).

```
select a.id_angajat, a.NumeAngajat, a.PrenumeAngajat
```

```
from angajat a
```

```
where not exists (
```

```
select r.id_rol
```

```
from rol_angajat r
```

```

where not exists (
    select *
    from (
        select c.id_rol
        from angajat a2
        join rol_angajat c on a2.id_rol = c.id_rol
        where a2.id_angajat = a.id_angajat
        union all
        select null
        from dual
    )
    minus
    select c.id_rol
    from rol_angajat c
)
);

```

```

select a.id_angajat, a.NumeAngajat, a.PrenumeAngajat
from angajat a
where not exists (
    select r.id_rol
    from rol_angajat r
    where not exists (
        select *
        from (
            select c.id_rol
            from angajat a2
            join rol_angajat c on a2.id_rol = c.id_rol
            where a2.id_angajat = a.id_angajat
            union all
            select null
            from dual
        )
        minus
        select c.id_rol
        from rol_angajat c
    )
);

```

ID_ANGAJAT	NUMEANGAJAT	PRENUMEANGAJAT
1	6 Ionescu	Anastasia
2	7 Florescu	Letitia
3	8 Stanescu	Stefan
4	9 Dumitache	Rafael
5	10 Teodorescu	Eleonora

Exercitiul 3 (top-n):

Afiseaza numele si prenumele clientilor, impreuna cu numarul total de recenzii primite, in ordine descrescatoare, limitat la primele 3 inregistrari. Deci un top 3 al clientilor cu cele mai multe recenzii postate.

```

select *
from (
    select c.NumeClient, c.PrenumeClient, count(r.id_recenzie) as NumarRecenzii
    from client c
    left join recenzie r on c.id_client = r.id_client
    group by c.NumeClient, c.PrenumeClient
    order by count(r.id_recenzie) desc
) where rownum <= 3;

```

```

3 select *
4   from (
5     select c.NumeClient, c.PrenumeClient, count(r.id_recenzie) as NumarRecenzii
6       from client c
7       left join recenzie r on c.id_client = r.id_client
8       group by c.NumeClient, c.PrenumeClient
9       order by count(r.id_recenzie) desc
10 ) where rownum <= 3;

```

	NUMECLIENT	PRENUMECLIENT	NUMARRECENZII
1	Feurdean	Mihai	2
2	Nica	Ana	1
3	Popescu	Ioana	1

Cerinta 16:

Optimizarea unei cereri, aplicând regulile de optimizare ce derivă din proprietățile operatorilor algebrei relaționale. Cererea va fi exprimată prin expresie algebraică, arbore algebraic și limbaj (SQL), atât anterior cât și ulterior optimizării.

Neoptimizat:

- Afisarea clientilor care au plasat cel putin o comanda si au adresa in Bucuresti:

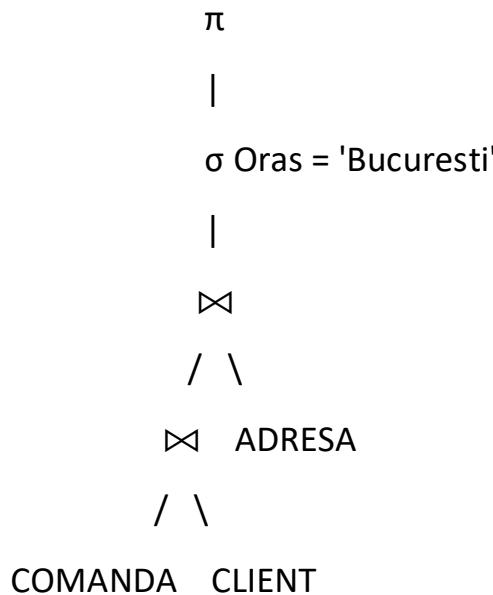
```
select c.numeclient, c.prenumeclient  
from comanda co  
join client c on co.id_client=c.id_client  
join adresa a on a.id_adresa=c.id_adresa  
where a.oras='Bucuresti';
```

- Expresie algebraica:

$$\pi_{\text{numeclient}, \text{prenumeclient}} (\sigma_{\text{oras} = \text{'Bucuresti'}} (\text{COMANDA} \bowtie \text{CLIENT} \bowtie \text{ADRESA}))$$

)

- Arboare algebric:



Optimizat:

```

select c.numeclient, c.prenumeclient
from comanda co
join client c on co.id_client=c.id_client
join (select * from adresa where oras='Bucuresti') a on a.id_adresa=c.id_adresa;
  
```

- Expresie algebraica:

$$\begin{aligned}
 & \pi \text{ numeclient, prenumeclient (} \\
 & \text{ COMANDA} \bowtie (\text{CLIENT} \bowtie (\sigma \text{ oras} = \text{'Bucuresti'} (\text{ADRESA}))) \\
 & \text{)}
 \end{aligned}$$

- Astfel, prin mutarea filtrului mai aproape de operatia de imperechere se reduce numarul de inregistrari prelucrate inainte de aplicarea filtrului.

- Arboare algebric:

```

    π
    |
    ╲
    / \
    σ oras = 'Bucuresti' (ADRESA)
    / \
    COMANDA CLIENT
  
```

Cerinta 17:

a. Realizarea normalizării BCNF [, FN4, FN5].

b. Aplicarea denormalizării, justificând necesitatea acesteia.

ALTERNATIVĂ: alegerea unor relații/ join-uri din model și reprezentarea acestora într-o bază de date NoSql (MongoDb, Cassandra etc.)

Forma normală Boyce-Codd (BCNF):

Forma normală Boyce-Codd se bazează pe dependențele funcționale care iau în considerare toate cheile candidat dintr-o relație. Pentru relațiile cu o singură cheie candidat, formele FN3 și BCNF sunt la fel.

In cazul tabelei „FURNIZOR” avem două atribută, „TelefonFurnizor” și „AdresaFurnizor”, care sunt unice pentru fiecare înregistrare. Acest lucru indică existența unor dependențe funcționale, între „id_furnizor” și „TelefonFurnizor” și între „id_furnizor” și „AdresaFurnizor”. Astfel, tabelul „FURNIZOR” nu se află în forma normală BCNF.

Pentru a aduce tabela „FURNIZOR” în forma normală BCNF, ar trebui să separem atributele „TelefonFurnizor” și „AdresaFurnizor” în două tabele noi

separate. Astfel, in loc de tabela „FURNIZOR” cu atributele „id_furnizor”, „NumeFurnizor”, „AdresaFurnizor”, „TelefonFurnizor”, avem urmatoarele tabele:

- FURNIZOR (id_furnizor (PK), NumeFurnizor)
- ADRESA_FURNIZOR (id_furnizor (FK), AdresaFurnizor (PK))
- TELEFON_FURNIZOR (id_furnizor (FK), TelefonFurnizor (PK))

Aceasta este structura finala pentru a aduce tabela „FURNIZOR” in forma normala BCNF, descompunand dependentele functionale si pastrand integritatea si corectitudinea datelor.

Forma normala 4 (FN4):

FN4 elimina redundantele datorate relatiilor many-to-many, adica datorate dependentei multiple. O relatie este in FN4 daca si numai daca este in BCNF si nu contine relatii many-to-many independente.

Forma normala 5 (FN5, forma normala proiectie-uniune):

O relatie este in FN5 daca si numai daca orice dependenta de uniune a relatiei este o consecinta a unei chei candidat a relatiei. Orice relatie care este in FN5 este si in FN4. Pentru a preciza daca o relatie este in FN5, trebuie sa cunoastem cheile candidate si toate dependentele de uniune ale relatiei. Aducerea in FN5 inseamna eliminarea join dependentelor.

Denormalizare:

Denormalizarea este procesul invers procesului de normalizare si are rolul de a realiza executarea mai rapida a interogarilor prin introducerea redundantei. Astfel, se pune accentul pe rapiditatea analizei si se scade numarul de tabele.

In baza noastră de date, denormalizarea subliniaza conceptul ca, plasand toate datele intr-un singur loc, ar putea elimina necesitatea cautarii acestor fisiere multiple pentru a colecta aceste date.

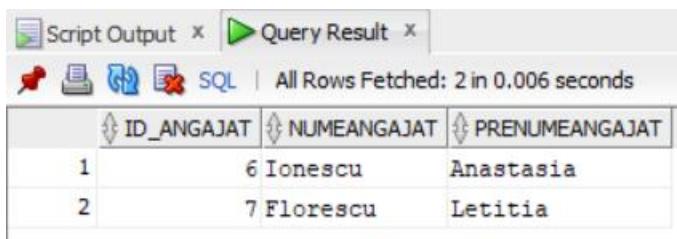
Cerinta 19:

Optimizarea a două cereri utilizând indexare.

Cererea 1: Selectarea tuturor angajatilor care au un anumit rol.

```
select id_angajat, NumeAngajat, PrenumeAngajat  
from angajat  
where id_rol =1;
```

```
select id_angajat, NumeAngajat, PrenumeAngajat  
from angajat  
where id_rol =1;
```

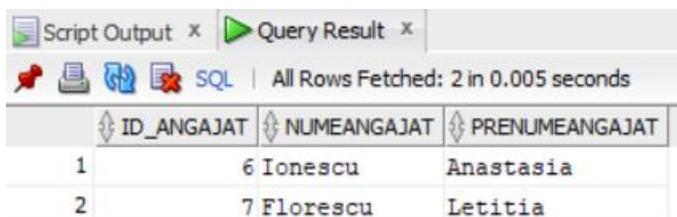


ID_ANGAJAT	NUMEANGAJAT	PRENUMEANGAJAT
1	6 Ionescu	Anastasia
2	7 Florescu	Letitia

Pentru a optimiza aceasta cerere, putem crea un index pe coloana „id_rol” din tabela „ANGAJAT”. Acest index va accelera cauterile pe baza valorilor din coloana „id_rol”, permitand o cautare mai rapida si eficienta a angajatilor cu rolul specificat.

```
create index idx_angajat_id_rol on angajat (id_rol);
```

```
Index IDX_ANAJAT_ID_ROL created.
```



ID_ANGAJAT	NUMEANGAJAT	PRENUMEANGAJAT
1	6 Ionescu	Anastasia
2	7 Florescu	Letitia

Cererea 2: Selectarea tuturor produselor dintr-o anumita categorie.

```
select id_produs, NumeProdus  
from produs  
where id_categprod = 34;
```

```
select id_produs, NumeProdus  
from produs  
where id_categprod = 34;
```

The screenshot shows the Oracle SQL Developer interface. The 'Query Result' tab is active, displaying the results of the executed SQL query. The results are presented in a table with two columns: 'ID_PRODUS' and 'NUMEPRODUS'. The data rows are: 1, 55 Trandafir englezesc; 2, 56 Cupressus sempervirens; 3, 57 Bujor.

ID_PRODUS	NUMEPRODUS
1	55 Trandafir englezesc
2	56 Cupressus sempervirens
3	57 Bujor

Pentru a optimiza aceasta cerere, putem crea un index pe coloana „id_categprod” din tabela „PRODUS”. Acest index va accelera cautarile pe baza valorilor din coloana „id_categprod”, permitand o cautare mai rapida si eficienta a produselor din categoria specificata.

```
create index idx_produs_id_categprod on produs (id_categprod);
```

```
Index IDX_PRODUS_ID_CATEGPROD created.
```

The screenshot shows the Oracle SQL Developer interface. The 'Query Result' tab is active, displaying the results of the executed SQL query. The results are presented in a table with two columns: 'ID_PRODUS' and 'NUMEPRODUS'. The data rows are: 1, 55 Trandafir englezesc; 2, 56 Cupressus sempervirens; 3, 57 Bujor.

ID_PRODUS	NUMEPRODUS
1	55 Trandafir englezesc
2	56 Cupressus sempervirens
3	57 Bujor

Prin crearea acestor indexari, interogarile respective vor beneficia de o performanta imbunatatita, deoarece cautarile vor fi eficientizate prin utilizarea indexului in locul scanarii complete a tabelelor.