

Baza de date

COFETARIE



1. Modelul ER entitate – legatura. Prezentarea entitatilor si a legaturilor dintre acestea, impreuna cu cardinalitatea lor.

Am mers pe principiul urmator:

O cofetarie are angajati, care au functie. Dar angajatii pregatesc produse, iar produsele contin ingrediente. Clientii comanda produsele, iar apoi se creeaza factura aferenta comenzii.

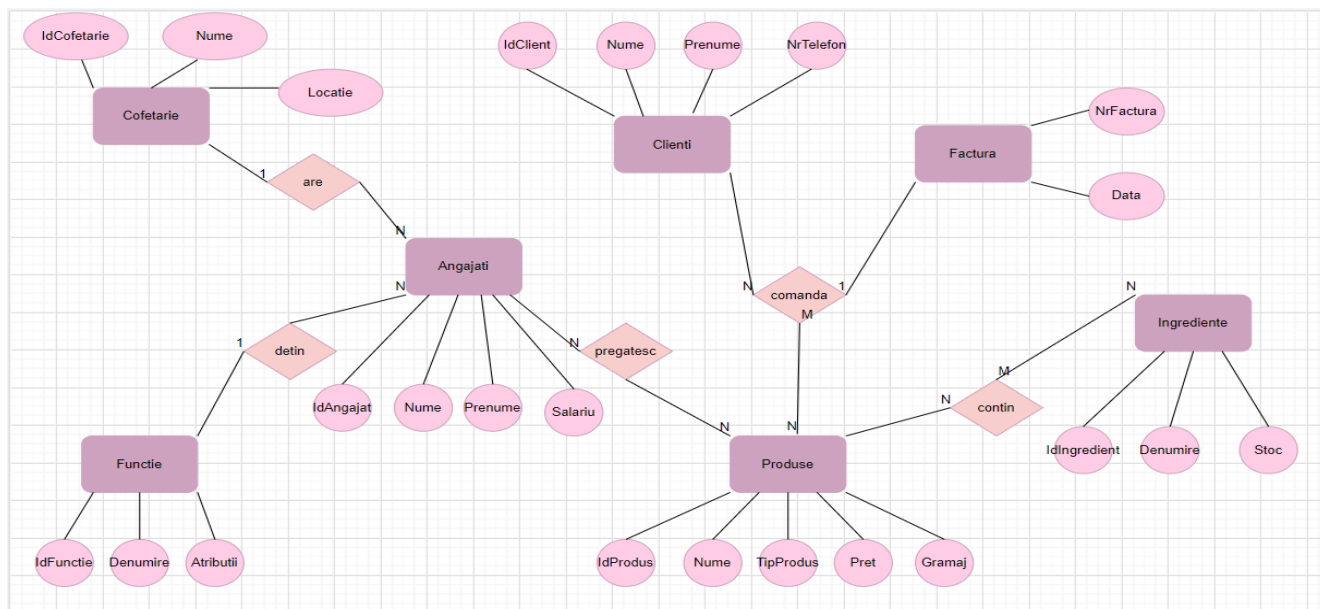
Vom avea legatura multi la multi intre Comanda si Produse pentru ca o comanda poate include mai mult de un produs. De asemenea, un produs poate aparea in mai multe comenzi.

Legatura multi la multi vom mai avea si intre Produse si Ingrediente, cu relatia Contin pentru ca un Produs contine mai multe Ingrediente, in mod logic.

Aceste legaturi multi la multi inseamna ca mai multor tupluri din relatia1 ii corespund mai multe tupluri ale relatiei2 si invers.

Acolo unde avem legaturi 1 la mai multi (1:N) inseamna ca unui tuplu din relatia1 ii corespund mai multe tupluri ale relatiei2 si invers. De exemplu, avem relatia Clienti – comanda – Produse, ceea ce inseamna ca un Client poate comanda mai multe Produse. Un alt exemplu poate fi Cofetarie – are – Angajati care presupune ca o cofetarie are mai multi angajati.

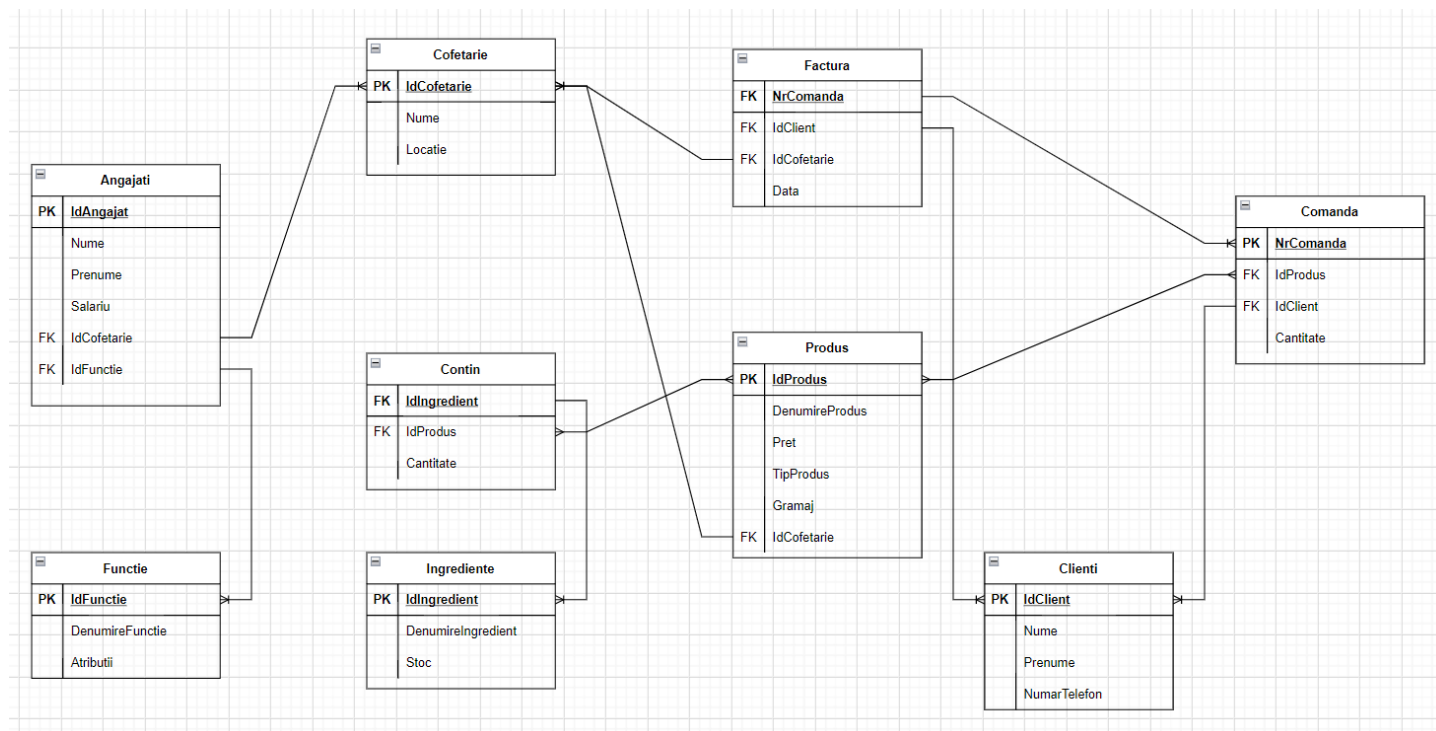
În acest model avem entitățile Cofetarie, Angajati, Functie, Clienti, Produse, Ingrediente și Factura.



2. Schema relationala (stabilirea relațiilor, atributelor, cheilor primare, constrangerilor, legăturilor dintre relații - chei externe)

Schema relationala este prezentată mai jos:

Am stabilit relațiile între toate tabelele, am definitivat cheile primare din fiecare tabel, precum și cheile externe.



Chei primare (primary keys):

Angajati -> PK_IdAngajat

Functie -> PK_IdFunctie

Cofetarie -> PK_IdCofetarie

Produs -> PK_IdProdus

Ingrediente -> PK_IdIngredient

Clienti -> PK_IdClient

Comanda -> PK_NrComanda

Chei externe (foreign keys):

Angajati -> FK_IdCofetarie,
FK_IdFunctie

Factura -> FK_IdClient,
FK_IdCofetarie, FK_NrComanda

Produs -> FK_IdCofetarie

Contin -> FK_IdIngredient,
FK_IdProdus

Comanda -> FK_IdProdus,
FK_IdClient

3. Implementarea modelului relational folosind SQL – crearea tabelelor, constrangerilor + Operatii de manipulare a relatiilor/tabelelor – inserare, modificare, stergere.

a. Fiecare tabel va contine cel putin 10 inregistrari.

b. Se vor da 4 exemple de operatii de update si 4 operatii delete.

Crearea tabelelor, alaturi de constrangeri:

```
create table Cofetarie
(
    IdCofetarie int primary key,
    Nume varchar(30) not null,
    Locatie varchar(30) not null
)
```

```
create table Ingrediente
(
    IdIngredient int primary key,
    DenumireIngredient varchar(30),
    Stoc varchar(30)
)
```

```
create table Functie
(
    IdFunctie int primary key,
    DenumireFunctie varchar(30),
    Atributii varchar(100)
)
```

```
create table Clienti
(
    IdClient int primary key,
    Nume varchar(30),
    Prenume varchar(30),
    NumarTelefon varchar(11)
)
```

```
create table Angajati
(
    IdAngajat int primary key,
    Nume varchar(30) not null,
    Prenume varchar(30) not null,
    Salariu int,
    IdCofetarie int,
    IdFunctie int
)
ALTER TABLE Angajati ADD CONSTRAINT
FK_IdCofetarie FOREIGN KEY (IdCofetarie) REFERENCES Cofetarie(IdCofetarie) on delete cascade on update
cascade
ALTER TABLE Angajati ADD CONSTRAINT
FK_IdFunctie FOREIGN KEY (IdFunctie) REFERENCES Functie(IdFunctie) on delete cascade on update cascade
```

```
create table Produs
(
    IdProdus int primary key,
    DenumireProdus varchar(30) not null,
    TipProdus varchar(30) not null,
    Pret int,
    Gramaj varchar(30) not null,
    IdCofetarie int
)
ALTER TABLE Produs ADD CONSTRAINT
FK_IdCofetarie FOREIGN KEY (IdCofetarie) REFERENCES Cofetarie(IdCofetarie) on delete cascade on update
cascade
```

```
create table Factura
(
    NrComanda int,
    IdClient int,
    IdCofetarie int,
    Data date
)
ALTER TABLE Factura ADD CONSTRAINT
FK_NrComanda FOREIGN KEY (NrComanda) REFERENCES Comanda(NrComanda) on delete cascade on update
cascade
ALTER TABLE Factura ADD CONSTRAINT
FK_IdClient FOREIGN KEY (IdClient) REFERENCES Clienti(IdClient) on delete cascade on update cascade
ALTER TABLE Factura ADD CONSTRAINT
FK_IdCofetarie FOREIGN KEY (IdCofetarie) REFERENCES Clienti(IdClient) on delete cascade on update cascade
```

```
create table Contin
(
    IdIngredient int,
    IdProdus int,
    Cantitate varchar(30)
)
ALTER TABLE Contin ADD CONSTRAINT
FK_IdIngredient FOREIGN KEY (IdIngredient) REFERENCES Ingrediente(IdIngredient) on delete cascade on
update cascade
ALTER TABLE Contin ADD CONSTRAINT
FK_IdProdus FOREIGN KEY (IdProdus) REFERENCES Produs(IdProdus) on delete cascade on update cascade
```

```
create table Comanda
(
    NrComanda int primary key,
    IdProdus int,
    IdClient int,
    Cantitate int
)
ALTER TABLE Comanda ADD CONSTRAINT
FK_IdProdus FOREIGN KEY (IdProdus) REFERENCES Produs(IdProdus) on delete cascade on update cascade
ALTER TABLE Comanda ADD CONSTRAINT
FK_IdClient FOREIGN KEY (IdClient) REFERENCES Clienti(IdClient) on delete cascade on update cascade
```

a. Inserari in tabele

Cofetarie

```
insert into Cofetarie values (1, 'Vivien', 'Craiova');
insert into Cofetarie values (2, 'Amandina', 'Filiasi');
insert into Cofetarie values (3, 'RIO', 'Craiova');
insert into Cofetarie values (4, 'Arkadia', 'Brasov');
insert into Cofetarie values (5, 'Sweet Escape', 'Sibiu');
insert into Cofetarie values (6, 'Eugenia', 'Bucuresti');
insert into Cofetarie values (7, 'Bella', 'Arad');
insert into Cofetarie values (8, 'Cakes', 'Timisoara');
insert into Cofetarie values (9, 'Boromir', 'Podari');
insert into Cofetarie values (10, 'Dessert Time', 'Constanta');
```

Angajati

```
insert into Angajati values (1, 'Popa', 'Robert', 5000, 1, 3);  
insert into Angajati values (2, 'Andrei', 'Marius', 8000, 6, 10);  
insert into Angajati values (3, 'Frunzulita', 'Mihnea', 4000, 1, 1);  
insert into Angajati values (4, 'Ionescu', 'Rica', 2500, 7, 2);  
insert into Angajati values (11, 'Crenguta', 'Mihai', 3000, 2, 4);  
insert into Angajati values (12, 'Dumitru', 'Daniel', 4500, 1, 4);  
insert into Angajati values (13, 'Vanghelie', 'Marian', 4300, 5, 4);  
insert into Angajati values (5, 'Patrascu', 'Niculina', 5000, 9, 5);  
insert into Angajati values (6, 'Petrescu', 'Liviu', 9000, 10, 6);  
insert into Angajati values (7, 'Hora', 'George', 1000, 3, 7);  
insert into Angajati values (8, 'Radu', 'Adrian', 4000, 4, 8);  
insert into Angajati values (9, 'Popa', 'Beti', 10000, 4, 9);  
insert into Angajati values (14, 'Popa', 'Ana', 10500, 3, 9);  
insert into Angajati values (10, 'Frunzulita', 'Ovi', 5000, 2, 10);  
insert into Angajati values (15, 'Veverita', 'Manole', 200000, 9, 10);  
insert into Angajati values (16, 'Pensula', 'Ramona', 5670, 5, 4);  
insert into Angajati values (17, 'Albastrea', 'Mirela', 7612, 7, 6);
```

Contin

```
insert into Contin values (1, 1, '50g');  
insert into Contin values (1, 7, '35g');  
insert into Contin values (1, 9, '90g');  
insert into Contin values (7, 1, '100g');  
insert into Contin values (2, 2, '33g');  
insert into Contin values (3, 3, '50g');  
insert into Contin values (4, 4, '3g');
```

```
insert into Contin values (4, 4, '3g');  
insert into Contin values (5, 5, '1kg');  
insert into contin values (12, 5, '15g');  
insert into Contin values (6, 6, '50g');  
insert into Contin values (7, 9, '90g');  
insert into Contin values (7, 7, '200g');  
insert into Contin values (8, 7, '50g');  
insert into Contin values (11, 10, '25g');  
insert into contin values (12, 12, '30g');
```

Produs

```
insert into Produs values (1, 'Amandina', 'Prajituri',6, '100g', 1);
insert into Produs values (11, 'Tort', 'Torturi',30, '300g', 1);
insert into Produs values (12, 'Chec', 'Prajituri',12, '350g', 1);
insert into Produs values (13, 'Pandispan cu visine', 'Prajituri',21, '102g', 1);
insert into Produs values (2, 'Clatita', 'Deserturi',3, '30g', 1);
insert into Produs values (3, 'Placinta', 'Foietaje',4.5, '75g', 2);
insert into Produs values (19, 'Carpati', 'Prajituri',5.5, '70g', 2);
insert into Produs values (15, 'Lava Cake', 'Prajituri',9, '100g', 2);
insert into Produs values (16, 'Poale-n brau', 'Foietaje',3.5, '60g', 2);
insert into Produs values (17, 'Briose', 'Prajituri',2.5, '85g', 2);
insert into Produs values (18, 'Negrese', 'Prajituri',5, '30g', 2);
insert into Produs values (4, 'Ecler', 'Prajituri',4.5, '75g', 2);
insert into Produs values (5, 'Cozonac', 'Prajitura',9, '1.5kg', 3);
insert into Produs values (6, 'Saratele', 'Foietaje',1.5, '15g', 3);
insert into Produs values (7, 'Tort', 'Torturi',35, '5kg', 3);
insert into Produs values (8, 'Strudel', 'Foietaje', 3, '50g', 1);
insert into Produs values (9, 'Tiramisu', 'Prajituri', 12, '100g', 5);
insert into Produs values (10, 'Fursecuri cu ovaz', 'Fursecuri', 4, '25g', 7);
insert into Produs values (14, 'Rulada', 'Prajituri', 7, '76g', 1);
insert into Produs (IdProdus, DenumireProdus, TipProdus, Gramaj, IdCofetarie) values
(20,'Cornulete', 'Foietaje', '15g', 9)
```


Funcție

```
insert into Funcție values (1, 'Ajutor de cofetar', 'ajuta cofetarul');  
insert into Funcție values (2, 'Om de serviciu', 'mentine curatenia');  
insert into Funcție values (3, 'Cofetar', 'pregateste delicatesele');  
insert into Funcție values (4, 'Livrator', 'livreaza comenzile');  
insert into Funcție values (5, 'Chelner', 'aduce delicatesele la masa');  
insert into Funcție values (6, 'Picolo', 'ajuta chelnerul');  
insert into Funcție values (7, 'Manipulator de marfa', 'aduce  
produse/ingrediente');  
insert into Funcție values (8, 'Casier', 'incaseaza banii');  
insert into Funcție values (9, 'Manager', 'pregateste bonurile de masa');  
insert into Funcție values (10, 'Patron', 'conduce afacerea');  
insert into Funcție values (11, 'Contabil', 'tine evidenta  
produselor/banilor');
```

Ingrediente

```
insert into Ingrediente values (1, 'Cacao', '15 bucati');  
insert into Ingrediente values (2, 'Dulceata de visine', '3 bucati');  
insert into Ingrediente values (3, 'Branza', '100 de bucati');  
insert into Ingrediente values (4, 'Crema de vanilie', '99 de bucati');  
insert into Ingrediente values (5, 'Faina', '8 bucati');  
insert into Ingrediente values (6, 'Ou', '13 bucati');  
insert into Ingrediente values (7, 'Frisca', '9 bucati');  
insert into Ingrediente values (8, 'Mere', '54 de bucati');  
insert into Ingrediente values (9, 'Ness', '32 de bucati');  
insert into Ingrediente values (10, 'Ovaz', '160 de bucati');  
insert into Ingrediente values (11, 'Lapte', '1000 de bucati');  
insert into Ingrediente values (12, 'Nuci', '20 de bucati');
```

Comanda

```
insert into Comanda values (200, 1, 6, 10);  
insert into Comanda values (562, 2, 8, 15);  
insert into Comanda values (13, 1, 2, 20);  
insert into Comanda values (250, 3, 5, 11);  
insert into Comanda values (578, 4, 6, 14);  
insert into Comanda values (999, 7, 9, 21);  
insert into Comanda values (700, 10, 7, 29);  
insert into Comanda values (372, 2, 1, 34);
```

```
insert into Comanda values (110, 9, 1, 45);  
insert into Comanda values (201, 4, 10, 3);  
insert into Comanda values (202, 3, 8, 9);  
insert into comanda values (207, 3, 10, 2);  
insert into comanda values (209, 16, 9, 83);  
insert into comanda values (210, 10, 4, 340);  
insert into comanda values (222, 14, 5, 29);  
insert into comanda values (204, 15, 5, 200);
```

Cienti

```
insert into Cienti values ('1', 'Pintilie', 'Dragos',  
0722663451);  
insert into Cienti values ('2', 'Oprea', 'Sonia',  
0745728039);  
insert into Cienti values ('3', 'Popa', 'Marian',  
0733449999);  
insert into Cienti values ('4', 'Frunza', 'George',  
0795836039);  
insert into Cienti values ('5', 'Mos', 'Craciun',  
0399100100);  
insert into Cienti values ('6', 'Udrica',  
'Marioara', 0735128039);
```

```
insert into Cienti values ('7', 'Florica',  
'Mihaela', 0702030405);  
insert into Cienti values ('8', 'Barbu', 'Ionut',  
0765555000);  
insert into Cienti values ('9', 'Popescu', 'Ovidiu',  
0799665523);  
insert into Cienti values ('10', 'Mihai', 'Viteazu',  
0732188444);  
insert into Cienti values ('11', 'Grigore', 'Bobel',  
0732652444);
```

Factura

```
insert into Factura values (200, 6, 1, '07/04/2021');  
insert into Factura values (201, 10, 2, '08/04/2021');  
insert into Factura values (207, 10, 3, '08/04/2021');  
insert into Factura values (202, 8, 4, '07/04/2021');  
insert into Factura values (562, 8, 4, '10/10/2021');  
insert into Factura values (250, 5, 5, '26/05/2021');
```

```
insert into Factura values (13, 2, 6, '14/10/2021');  
insert into Factura values (578, 6, 7, '22/09/2021');  
insert into Factura values (999, 9, 8, '18/01/2021');  
insert into Factura values (700, 7, 9, '15/03/2021');  
insert into Factura values (372, 1, 10, '28/02/2021');  
insert into Factura values (110, 1, 11, '30/08/2021');  
insert into Factura values (209, 9, 8, '30/08/2021');  
insert into Factura values (210, 4, 8, '30/08/2021');  
insert into Factura values (222, 5, 8, '30/08/2021');  
insert into Factura values (204, 5, 1, '30/08/2021');
```

b. Operatii update + delete (4)

Update

```
update Angajati set Salariu = 9000 where IdAngajat = 6;  
update Angajati set Salariu = 8000 where IdAngajat = 14;  
update Angajati set IdFuncție = 10 where IdAngajat = 9  
update Factura set Data = '07/04/2021' where NrComanda = 201
```

Delete

```
delete from angajati where idangajat = 17;  
delete from factura where idcofetarie = 11;  
delete from produs where denumireprodus = 'Negrese';  
delete from ingrediente where idingredient = 13;
```

4. Interogari –Se vor da exemple cu minim 20 de interogari

a. Minim 5 interogari cu operatii de jonctiune (inner join+outer join)

b. Minim 5 interogari care sa contina functii de agregare si Group by

c. Minim 7 exemple de subinterogari (IN , ANY, Exists, etc)

d. Minim 3 vederi.

a.

1) Inner join

➤ Afisam produsele care contin ingredientele cacao sau frisca

```
327 --afisarea produselor care contin ingredientele cacao sau frisca - 1 si 7
328 select distinct produs.idprodus, produs.denumireprodus
329 from contin inner join produs on produs.idprodus = contin.idprodus
330 where contin.idingredient = 7 or contin.idingredient = 1
```

	idprodus [PK] integer	denumireprodus character varying (30)
1	1	Amandina
2	7	Tort
3	9	Tiramisu

➤ Afisam numarul comenzii, numele clientilor si data la care s-a efectuat fiecare comanda

```
402 --afisarea nr comenzii, numele clientilor,
403 --data la care s-a efectuat comanda
404 SELECT factura.nrcomanda, clienti.numa, clienti.prenume, factura.data
405 FROM factura
406 INNER JOIN clienti ON factura.idclient = clienti.idclient;
```

	nrcomanda integer	numa character varying (30)	prenume character varying (30)	data date
1	200	Udrica	Marioara	2021-04-07
2	202	Barbu	Ionut	2021-04-07
3	562	Barbu	Ionut	2021-10-10
4	250	Mos	Craciun	2021-05-26
5	13	Oprea	Sonia	2021-10-14
6	578	Udrica	Marioara	2021-09-22
7	999	Popescu	Ovidiu	2021-01-18
8	700	Florica	Mihaela	2021-03-15
9	372	Pintilie	Dragos	2021-02-28
10	201	Mihai	Viteazu	2021-04-07
11	207	Mihai	Viteazu	2021-04-08
12	209	Popescu	Ovidiu	2021-08-30
13	210	Frunza	George	2021-08-30
14	222	Mos	Craciun	2021-08-30
15	204	Mos	Craciun	2021-08-30

- Afisam numele clientilor si numarul comenzilor, chiar daca exista clienti care nu au dat inca o comanda

```
408 --afisam numele clientilor si nr comenzii fiecaruia
409 --chiar daca unii clienti nu au dat inca o comanda
410 SELECT clienti.numa, clienti.prenume, factura.nrcomanda
411 FROM clienti
412 LEFT JOIN factura ON clienti.idclient = factura.idclient
413 ORDER BY factura.nrcomanda
```

	nume character varying (30)	prenume character varying (30)	nrcomanda integer
1	Oprea	Sonia	13
2	Udrica	Marioara	200
3	Mihai	Viteazu	201
4	Barbu	Ionut	202
5	Mos	Craciun	204
6	Mihai	Viteazu	207
7	Popescu	Ovidiu	209
8	Frunza	George	210
9	Mos	Craciun	222
10	Mos	Craciun	250
11	Pintilie	Dragos	372
12	Barbu	Ionut	562
13	Udrica	Marioara	578
14	Florica	Mihaela	700
15	Popescu	Ovidiu	999
16	Grigore	Bobel	[null]
17	Popa	Marian	[null]

- Numaram comenzile date de la fiecare cofetarie

```
428 --numaram comenzile facute de la fiecare cofetarie
429 --si grupam dupa numele cofetariei
430 SELECT cofetarie.numa, COUNT(factura.nrcomanda) AS nr_comenzi FROM factura
431 LEFT JOIN cofetarie ON factura.idcofetarie = cofetarie.idcofetarie
432 GROUP BY cofetarie.numa
```

	nume character varying (30)	nr_comenzi bigint
1	RIO	1
2	Amandina	1
3	Eugenia	1
4	Vivien	2
5	Arkadia	2
6	Dessert Time	1
7	Sweet Escape	1
8	Cakes	4
9	Boromir	1
10	Bella	1

- Afisam numele clientilor care au comandat + numarul comenzilor fiecaruia si cantitatea comandata

```
505 --afisam clientii si nr comenzilor fiecaruia
506 --ordonam crescator
507 select clienti.idclient, clienti.numa, comanda.nrcomanda, cantitate
508 from clienti
509 inner join comanda ON clienti.idclient = comanda.idclient order by comanda.nrcomanda asc
```

	idclient integer	numa character varying (30)	nrcomanda integer	cantitate integer
1	2	Oprea	13	20
2	1	Pintilie	110	45
3	6	Udrica	200	10
4	10	Mihai	201	3
5	8	Barbu	202	9
6	5	Mos	204	200
7	10	Mihai	207	2
8	9	Popescu	209	83
9	4	Frunza	210	340
10	5	Mos	222	29
11	5	Mos	250	11
12	1	Pintilie	372	34
13	8	Barbu	562	15
14	6	Udrica	578	14
15	7	Florica	700	29
16	9	Popescu	999	21

- Afisam id-ul clientilor care au comandat fursecuri cu ovaz (id 10)

```
519 --afisarea clientilor care au comandat fursecuri cu ovaz
520 select distinct clienti.idclient, clienti.numa
521 from clienti inner join comanda on clienti.idclient = comanda.idclient
522 where comanda.idprodus = 10
```

	idclient [PK] integer	numa character varying (30)
1	4	Frunza
2	7	Florica

- Afisam produsele care contin ingredientele faina sau nuci

```
524 --afisarea produselor care contin ingredientele faina sau nuci
525 select distinct produs.idprodus, produs.denumireprodus
526 from contin inner join produs on produs.idprodus = contin.idprodus
527 where contin.idingredient = 5 or contin.idingredient = 12
```

	idprodus [PK] integer	denumireprodus character varying (30)
1	5	Cozonac
2	12	Chec

2. Outer join

- Afisam numele tuturor angajatilor si id-ul functiilor lor, chiar daca exista angajati care inca nu au functie

```
395 --numele tuturor angajatilor si functiile lor,
396 --chiar daca unii angajati nu au momentan functie
397 SELECT angajati.nume, functie.idfunctie
398 FROM angajati
399 FULL OUTER JOIN functie ON angajati.idangajat=functie.idfunctie
400 ORDER BY angajati.nume;
```

	nume character varying (30)	idfunctie integer
1	Andrei	2
2	Ariana	[null]
3	Crenguta	11
4	Dumitru	[null]
5	Frunzulita	3
6	Frunzulita	10
7	Hora	7
8	Ionescu	4
9	Patrascu	5
10	Pensula	[null]
11	Petrescu	6
12	Popa	1
13	Popa	9
14	Popa	[null]
15	Puya	[null]
16	Radu	8
17	Sticluta	[null]
18	Vanghelie	[null]
19	Veverita	[null]

- Afisam produsele de la cofetarii si pretul fiecarui produs, inclusiv produsele care nu au pret inca

```

415 --toate produsele de la toate cofetariile, precum si pretul fiecaruia
416 --ordonat crescator (apar si cele care nu au pret inca)
417 SELECT produs.denumireprodus, produs.pret, produs.idcofetarie
418 FROM produs
419 FULL OUTER JOIN cofetarie ON produs.idprodus=cofetarie.idcofetarie
420 ORDER BY produs.denumireprodus asc
...

```

	denumireprodus character varying (30)	pret integer	idcofetarie integer
1	Amandina	6	1
2	Briose	3	2
3	Carpati	6	2
4	Chec	12	1
5	Clatita	3	1
6	Cornulete	[null]	9
7	Cozonac	9	3
8	Ecler	5	2
9	Fursecuri cu ovaz	4	7
10	Lava Cake	9	2
11	Pandispan cu visine	21	1
12	Placinta	5	2
13	Poale-n brau	4	2
14	Rulada	7	1
15	Saratele	2	3
16	Strudel	3	1
17	Tiramisu	12	5
18	Tort	35	3
19	Tort	30	1

- Afisam numele tuturor angajatilor de la fiecare cofetarie si afisam inclusiv cofetariile fara angajati

```

532 --numele angajatilor care lucreaza la fiecare cofetarie
533 --chiar daca unelele cofetarii nu au inca angajati
534 SELECT a.idangajat, a.numa, c.idcofetarie
535 FROM angajati a full outer join cofetarie c on a.idcofetarie = c.idcofetarie
536 order by idcofetarie

```

	idangajat integer	numa character varying (30)	idcofetarie integer
1	1	Popa	1
2	12	Dumitru	1
3	3	Frunzulita	1
4	10	Frunzulita	2
5	18	Puya	2
6	20	Sticluta	2
7	19	Ariana	2
8	11	Crenguta	2
9	14	Popa	3
10	7	Hora	3
11	8	Radu	4
12	9	Popa	4
13	13	Vanghelie	5
14	16	Pensula	5
15	2	Andrei	6
16	4	Ionescu	7
17	[null]	[null]	8
18	15	Veverita	9
19	5	Patrascu	9
20	6	Petrescu	10

b. Functii de agregare si Group BY

1. COUNT

- Numaram comenzile facute de la fiecare cofetarie si grupam dupa numele cofetariei

```
428 --numaram comenzile facute de la fiecare cofetarie
429 --si grupam dupa numele cofetariei
430 SELECT cofetarie.nume, COUNT(factura.nrcomanda) AS nr_comenzi FROM factura
431 LEFT JOIN cofetarie ON factura.idcofetarie = cofetarie.idcofetarie
432 GROUP BY cofetarie.nume
```

	nume character varying (30)	nr_comenzi bigint
1	RIO	1
2	Amandina	1
3	Eugenia	1
4	Vivien	2
5	Arkadia	2
6	Dessert Time	1
7	Sweet Escape	1
8	Cakes	4
9	Boromir	1
10	Bella	1

- Afisam numarul total de angajati

```
511 --afiseaza numarul total de angajati
512 SELECT COUNT(*) FROM angajati
```

	count bigint
1	19

- Afisam cofetariile care au <=2 angajati

```
570 --afisarea cofetariilor care au mai mai putin sau egal cu 2 angajati (inclusiv 0)
571 select distinct cofetarie.nume
572 from cofetarie
573 where cofetarie.idcofetarie not in (select distinct angajati.idcofetarie
574                                     from angajati
575                                     group by angajati.idcofetarie
576                                     having count(*) >=2)
```

	nume character varying (30)
1	Eugenia
2	Dessert Time
3	Cakes
4	Bella

2. AVG

- Id-ul si numele angajatilor care castiga mai mult decat salariul mediu

```
459 --id-ul si numele angajatilor care castiga mai mult decat salariul mediu
460 select idangajat, nume, salariu
461 from angajati
462 where salariu > (SELECT AVG(salariu)
463                 FROM angajati)
464 Order by salariu
```

	idangajat [PK] integer	nume character varying (30)	salariu integer
1	15	Veverita	210000

- Afisam salariul mediu al tuturor angajatilor cu functie

```
466 --salariu mediu al tuturor angajatiilor cu functie de pana acum
467 SELECT AVG(salariu), idfunctie
468 FROM angajati
469 GROUP BY idfunctie
470 ORDER BY 1
```

	avg numeric	idfunctie integer
1	1000.0000000000000000	7
2	2500.0000000000000000	2
3	2750.0000000000000000	1
4	3954.0000000000000000	4
5	4000.0000000000000000	8
6	5000.0000000000000000	5
7	5000.0000000000000000	3
8	8000.0000000000000000	9
9	8350.0000000000000000	6
10	58500.0000000000000000	10

- Afisam salariul mediu doar daca este mai mare decat 4000

```
472 --salariul mediu doar daca acesta este mai mare decat 4000
473 SELECT avg(salariu)
474 FROM angajati
475 HAVING avg(salariu)>4000
```

	avg numeric
1	15866.842105263158

3. SUM

- Afisam suma salariilor pe functii in cadrul cofetariilor

```
477 --suma salariilor pe functii in cadrul cofetariilor
478 SELECT sum(salariu),idfunctie,idcofetarie
479 FROM angajati
480 GROUP BY idfunctie, idcofetarie
481 ORDER BY 2 --se ordoneaza dupa coloana 2, adica dupa idfunctie
```

	sum bigint	idfunctie integer	idcofetarie integer
1	1500	1	2
2	4000	1	1
3	2500	2	7
4	5000	3	1
5	5300	4	2
6	4500	4	1
7	9970	4	5
8	5000	5	9
9	7700	6	2
10	9000	6	10
11	1000	7	3
12	4000	8	4
13	8000	9	3
14	10000	10	4
15	9000	10	6
16	210000	10	9
17	5000	10	2

4. Sum, Max, Min, Avg

- Afisam salariul maxim, minim, media salariilor si suma salariilor de la cofetaria Vivien

```
390 --salariul maxim, salariul minim, media si suma salariilor de la cofetaria 1 (Vivien)
391 select sum(salariu), avg(salariu), max(salariu), min(salariu)
392 from angajati inner join cofetarie on angajati.idcofetarie = cofetarie.idcofetarie
393 where cofetarie.idcofetarie = 1
```

	sum bigint	avg numeric	max integer	min integer
1	13500	4500.0000000000000000	5000	4000

5. Group BY

```
528 --liniile cu nume duplicat sunt grupate sub acelasi nume
529 --si salariul lor corespunzator este suma salariului randurilor duplicate
530 select nume, sum(salariu)
531 from angajati
532 group by nume
```

	nume character varying (30)	sum bigint
1	Popa	23000
2	Andrei	9000
3	Radu	4000
4	Crenguta	3000
5	Frunzulita	9000
6	Patrascu	5000
7	Sticluta	7700
8	Dumitru	4500
9	Veverita	210000
10	Ionescu	2500
11	Petrescu	9000
12	Pensula	5670
13	Ariana	2300
14	Vanghelie	4300
15	Hora	1000
16	Puya	1500

- Afisam denumirea si gramajul pentru fiecare produs, grupat dupa idprodus

```
534 --denumirea si gramajul fiecarui produs
535 --grupat dupa idprodus
536 SELECT idprodus, denumireprodus, gramaj
537 FROM produs
538 Group BY idprodus
539 order by idprodus
```

	idprodus [PK] integer	denumireprodus character varying (30)	gramaj character varying (30)
1	1	Amandina	100g
2	2	Clatita	30g
3	3	Placinta	75g
4	4	Ecler	75g
5	5	Cozonac	1.5kg
6	6	Saratele	15g
7	7	Tort	5kg
8	8	Strudel	50g
9	9	Tiramisu	100g
10	10	Fursecuri cu ovaz	25g
11	11	Tort	300g
12	12	Chec	350g
13	13	Pandispan cu visine	102g
14	14	Rulada	76g
15	15	Lava Cake	100g
16	16	Poale-n brau	60g
17	17	Briose	85g
18	19	Carpati	70g
19	20	Cornulete	15g

- Cautam angajatii care au suma salariilor mai mica de 2000 si grupam dupa nume

```
541 --cauta angajatii care au suma salariilor < 2000
542 --si grupeaza dupa nume
543 SELECT nume, SUM(salariu) FROM angajati
544 GROUP BY nume
545 HAVING SUM(salariu)<2000;
```

	nume character varying (30)	sum bigint
1	Hora	1000
2	Puya	1500

- Numaram produsele de fiecare tip

```
547 --numaram cate produse avem din fiecare tip
548 SELECT tipprodus, COUNT(*) AS total_products
549 FROM produs
550 WHERE tipprodus IS NOT NULL
551 GROUP BY tipprodus
552 ORDER BY tipprodus
```

	tipprodus character varying (30)	total_products bigint
1	Deserturi	1
2	Foietaje	5
3	Fursecuri	1
4	Prajitura	1
5	Prajituri	9
6	Torturi	2

- Afisam salariul maxim de la fiecare cofetarie

```
554 --cel mai mare salariu de la fiecare cofetarie
555 SELECT distinct angajati.idcofetarie, MAX(salariu) AS highest_salary
556 FROM angajati
557 GROUP BY angajati.idcofetarie
558 order by idcofetarie
```

	idcofetarie integer	highest_salary integer
1	1	5000
2	2	7700
3	3	8000
4	4	10000
5	5	5670
6	6	9000
7	7	2500
8	9	210000
9	10	9000

c. Subinterogari

1. Any

- Afisam numele produselor care au gramajul de 50g

```
347 --afiseaza numele produseleor care au gramajul de 50g
348 SELECT denumireprodus |
349 FROM produs
350 where idprodus = any (SELECT idprodus
351                        FROM contin
352                        WHERE cantitate = '50g');
```

	denumireprodus character varying (30)
1	Amandina
2	Placinta
3	Saratele
4	Tort

- Afisam toate comenzile cu o cantitate mai mica decat cantitatea maxima de torturi comandata

```
354 --afiseaza toate comenzile cu o cantitate mai mica
355 --decat cantitatea maxima de torturi (id 7) comandata (21)
356 SELECT nrcomanda, idclient, cantitate
357 FROM comanda
358 WHERE cantitate < ANY (SELECT cantitate
359                        FROM comanda
360                        WHERE idprodus = 7)
361 order by cantitate asc
```

	nrcomanda [PK] integer	idclient integer	cantitate integer
1	207	10	2
2	201	10	3
3	202	8	9
4	200	6	10
5	250	5	11
6	578	6	14
7	562	8	15
8	13	2	20

- Afisam denumirea produsului unde cantitatea este mai mare de 21

```
363 --afiseaza denumirea produsului unde cantitatea > 21
364 SELECT denumireprodus
365 FROM produs
366 WHERE idprodus = ANY (SELECT idprodus
367                        FROM comanda
368                        WHERE cantitate > 21);
```

	denumireprodus character varying (30)
1	Clatita
2	Tiramisu
3	Fursecuri cu ovaz
4	Rulada
5	Lava Cake
6	Poale-n brau

2. All

- Afisam numele angajatilor cu salariul mai mare decat al tuturor angajatilor din cofetaria 2

```
340 --numele angajaților cu salariul mai mare decât
341 --al tuturor angajaților din cofetaria 2
342 SELECT Nume, Prenume, idcofetarie, salariu
343 FROM Angajati
344 WHERE Salariu > ALL (SELECT Salariu FROM Angajati WHERE IdCofetarie = 2);
```

	nume character varying (30)	prenume character varying (30)	idcofetarie integer	salariu integer
1	Popa	Beti	4	10000
2	Andrei	Marius	6	9000
3	Petrescu	Liviu	10	9000
4	Popa	Ana	3	8000
5	Veverita	Manole	9	210000

- Afisam salariile minime mai mari decat salariile minime pana la idfunctie = 5

```
370 --afiseaza salariile minime mai mari
371 --decat (salariile minime pana la idfunctie = 5)
372 SELECT idfunctie, MIN(salariu)
373 FROM angajati
374 GROUP BY idfunctie
375 HAVING MIN(salariu) > ALL (SELECT MIN(salariu)
376                             FROM angajati
377                             WHERE idfunctie < 5
378                             GROUP BY idfunctie)
```

	idfunctie integer	min integer
1	9	8000
2	6	7700

3. IN

- Afisam numele cofetariilor care prepara mai mult de 6 produse

```
332 --afisarea cofetariilor care prepara mai mult de 6 produse
333 select distinct nume
334 from cofetarie, produs
335 where cofetarie.idcofetarie in (select distinct produs.idcofetarie
336                                 from produs
337                                 group by produs.idcofetarie
338                                 having count(*) >=6)
```

	nume character varying (30)
1	Amandina
2	Vivien

- Cautam comenzile facute si id-urile produselor din comenzi, apoi afisam tot tabelul Produs in care se afla aceste comenzi, impreuna cu id-urile produselor din comenzi, denumirea produselor etc

```
412 --cauta comenzile facute si id-urile produselor din comenzi
413 --apoi afiseaza tot tabelul produs in care se afla aceste comenzi
414 --impreuna cu id-urile produselor din comenzi, denumirea produselor etc
415 SELECT * FROM produs
416 WHERE idprodus IN (SELECT idprodus FROM comanda);|
```

Data Output		Explain	Messages	Notifications		
	idprodus [PK] integer	denumireprodus character varying (30)	tipprodus character varying (30)	pret integer	gramaj character varying (30)	idcofetarie integer
1	1	Amandina	Prajituri	6	100g	1
2	2	Clatita	Deserturi	3	30g	1
3	3	Placinta	Foietaje	5	75g	2
4	4	Ecler	Prajituri	5	75g	2
5	7	Tort	Torturi	35	5kg	3
6	9	Tiramisu	Prajituri	12	100g	5
7	10	Fursecuri cu ovaz	Fursecuri	4	25g	7
8	14	Rulada	Prajituri	7	76g	1
9	15	Lava Cake	Prajituri	9	100g	2
10	16	Poale-n brau	Foietaje	4	60g	2

4. EXISTS

- Afisam numele clientilor care au comandat produse cu pretul de peste 30 lei

```
426 --numele clientilor care au comandat produse
427 --cu pretul de peste 30 de lei
428 SELECT *
429 FROM clienti
430 WHERE EXISTS (SELECT idprodus
431               FROM produs
432               WHERE produs.idprodus = clienti.idclient AND pret > 30)
```

Data Output		Explain	Messages	Notifications
	idclient [PK] integer	nume character varying (30)	prenume character varying (30)	numartelefon character varying (11)
1	7	Florica	Mihaela	702030405

- Afisam id-urile cofetariilor unde exista livratori (id = 4) si numele livratorilor

```
486 --id-urile cofetariilor unde exista livratori
487 --si numele livratorilor
488 SELECT angajati.nume, prenume, idcofetarie
489 FROM angajati
490 WHERE EXISTS (SELECT idfunctie
491               FROM functie
492               WHERE angajati.idfunctie = functie.idfunctie AND idfunctie = 4)
493 order by idcofetarie|
```

	nume character varying (30)	prenume character varying (30)	idcofetarie integer
1	Dumitru	Daniel	1
2	Crenguta	Mihai	2
3	Ariana	Grande	2
4	Vanghelie	Marian	5
5	Pensula	Ramona	5

d. Vederi

- Vizualizam angajatii care au acelasi nume de faimile si numaram la fiecare

```
560 --vizualizam angajatii care au acelasi nume de familie
561 --si numaram la fiecare
562 CREATE VIEW nrangajati AS SELECT nume, Count(*) as nrangajati
563                           FROM angajati
564                           group by nume
565                           order by nrangajati desc
566
567 select * from nrangajati
568 drop view nrangajati|
```

	nume character varying (30)	nrangajati bigint
1	Popa	3
2	Frunzulita	2
3	Radu	1
4	Crenguta	1
5	Patrascu	1
6	Sticluta	1
7	Dumitru	1
8	Veverita	1
9	Ionescu	1
10	Petrescu	1
11	Pensula	1
12	Ariana	1
13	Vanghelie	1
14	Hora	1
15	Puya	1
16	Andrei	1

- Vizualizam angajatii cu salariul mai mic de 3000, eligibili pentru marirea de salariu

```
473 --vedere in care se afiseaza angajatii cu salariul < 3000
474 --eligibili pt marirea de salariu
475 CREATE VIEW angajati_eligibili_marire_salariu(idangajat,nume, prenume )
476 AS ( SELECT idangajat, nume, prenume
477       DISTINCT FROM angajati
478       WHERE idangajat NOT IN (SELECT idangajat
479                                FROM angajati
480                                WHERE salariu > 3000)
481 )
482 drop view angajati_eligibili_marire_salariu
483 select * from angajati_eligibili_marire_salariu|
```

Data Output	Explain	Messages	Notifications
	idangajat integer	nume character varying (30)	prenume character varying (30)
1	4	Ionescu	Rica
2	7	Hora	George
3	11	Crenguta	Mihai
4	18	Puya	Kamelia-George
5	19	Ariana	Grande

- Vizualizam denumirea produselor care au pretul mai mare decat pretul mediu

```
434 --o vedere in care ni se afiseaza denumirea produselor
435 --care au pretul mai mare decat pretul mediu
436 CREATE VIEW produse_peste_pretul_mediu AS
437 SELECT denumireprodus, pret
438 FROM produs
439 WHERE pret > (SELECT AVG(pret) FROM produs);
440
441 select * from produse_peste_pretul_mediu|
```

	denumireprodus character varying (30)	pret integer
1	Tort	30
2	Chec	12
3	Pandispan cu visine	21
4	Tort	35
5	Tiramisu	12

➤ Vizualizam cofetariile cu locatiile in Craiova si Filiasi

```
297 --vizualizam cofetariile care au locatia in craiova si filiasi
298 create view locatii as
299 select * from cofetarie
300 where locatie in ('Craiova', 'Filiasi')
301
302 select * from locatii
```

	idcofetarie integer	nume character varying (30)	locatie character varying (30)
1	1	Vivien	Craiova
2	2	Amandina	Filiasi
3	3	RIO	Craiova

➤ Vizualizam suma fiecărei comenzi

```
290 --venitul fiecărei comenzi(suma) (pret x cantitate = suma platita)
291 create view venit as select produs.DenumireProdus produs, pret pret, comanda.Cantitate cantitate, (produs.Pret * comanda.Cantitate) venit, factura.Data Data
292 from produs, comanda, factura
293 where produs.IdProdus = comanda.IdProdus and comanda.NrComanda = factura.NrComanda
294 --drop view venit
295 select * from venit
```

	produs character varying (30)	pret integer	cantitate integer	venit integer	data date
1	Amandina	6	10	60	2021-04-07
2	Placinta	5	9	45	2021-04-07
3	Clatita	3	15	45	2021-10-10
4	Placinta	5	11	55	2021-05-26
5	Amandina	6	20	120	2021-10-14
6	Ecler	5	14	70	2021-09-22
7	Tort	35	21	735	2021-01-18
8	Fursecuri cu ovaz	4	29	116	2021-03-15
9	Clatita	3	34	102	2021-02-28
10	Ecler	5	3	15	2021-04-07
11	Placinta	5	2	10	2021-04-08
12	Poale-n brau	4	83	332	2021-08-30
13	Fursecuri cu ovaz	4	340	1360	2021-08-30
14	Rulada	7	29	203	2021-08-30
15	Lava Cake	9	200	1800	2021-08-30

5. Normalizarea relatiilor. Pe baza de date aleasa, se vor da exemple care nu respecta formele normale.

Forma normala 1 (FN1)

O relatie este in prima forma normala daca fiecarui atribut care o compune ii corespunde o valoare atomica.

Un produs contine mai multe ingrediente. De exemplu, cozonacul contine faina, nuci, zahar, oua, cacao...

Pentru a face FN1, am creat un tabel nou numit Contin care cuprinde relatia dintre produse si ingredientele continute.

Am avea un tabel despre produse initial:

IdProdus	DenumireProdus	Ingrediente	Cantitate
1	Cozonac	Faina, Nuci, Zahar, Oua	1kg, 15g, 200g, 30g



Am impartit tabelul in 3 tabele, astfel:

Produs

PK_IdProdus	DenumireProdus
1	Cozonac

Contin

FK_IdIngredient	FK_IdProdus
1	1
2	1
3	1
4	1

Ingrediente

PK_IdIngredient	DenumireIngredient
1	Faina
2	Nuci
3	Zahar
4	Oua

Forma normala 2 (FN2)

O relatie R este in a doua forma normala daca si numai daca:

- Relatia R este in FN1;
- Fiecare atribut care nu este cheie (nu face parte din cheia primara) este dependent de intreaga cheie primara.

Un angajat lucreaza la o anumita cofetarie. Puteam sa avem un tabel format din idangajat, nume, prenume, salariu, idcofetarie, numecofetarie, idfunctie, denumirefunctie, locatiecofetarie

Astfel am despartit acest tabel in 3 tabele:

Angajat(IdAngajat, Nume, Prenume, Salariu, IdCofetarie, IdFunctie)

Cofetarie(IdCofetarie, Nume, Locatie)

Functie(IdFunctie, DenumireFunctie, Atributii)

Tabel initial:

IdAngajat	Nume	IdCofetarie	NumeCofetarie	LocatieCofetarie	IdFunctie	DenumireFunctie	Atributii
1	Popa	1	Vivien	Craiova	1	Patron	conduce
2	Ion	2	Arkadia	Filiasi	2	Contabil	verifica
3	Mihai	1	Vivien	Craiova	3	Casier	incaseaza

Angajat

IdAngajat	Nume	IdCofetarie	IdFunctie
1	Popa	1	3
2	Ion	2	7
3	Mihai	1	5

Cofetarie

IdCofetarie	Nume	Locatie
1	Vivien	Craiova
2	Arkadia	Filiasi

Functie

IdFunctie	DenumireFunctie	Atributii
3	Patron	Conduce afacerea
5	Contabil	Verifica banii/comenzile
7	Casier	Incasseaza banii

Forma normala 3 (FN3)

O relatie R este in a treia forma normala daca si numai daca:

- Relatia R este in FN2;
- Fiecare atribut care nu este cheie (nu participa la o cheie) depinde direct de cheia primara

O relatie este in FN3 daca si numai daca fiecare atribut (coloana) care nu este cheie, depinde de cheie, de intreaga cheie si numai de cheie.

Fiecare angajat are o anumta functie. Puteam avea un tabel format din idangajat, nume, idfunctie, denumirefunctie

Astfel am creat 2 tabele:

Angajat(IdAngajat, nume, idfunctie)

Functie(IdFunctie, denumirefunctie)

Aici fiecare angajat are un id si un id al functiei aferente

Tabel initial:

IdAngajat	Nume	IdFunctie	DenumireFunctie
1	Popa	13	Picolo
2	Ion	15	Chelner

Angajati

IdAngajat	Nume	IdFunctie
1	Popa	13
2	Ion	15

Functie

IdFunctie	DenumireFunctie
13	Picolo
15	Chelner

Forma normala 4 (FN4)

FN4 elimina redundantele datorate relatiilor M:N (relatii multi la multi, dependente multiple).

Fiecare client comanda produse si se creeaza factura.

Deci am avea 2 tabele client si comanda, la care adaugam tabelul nou numit Factura

Client(IdClient, nume)

Comanda(NrComanda, IdClient)

Factura(NrComanda, IdClient, data)

Client

PK_IdClient	Nume
1	Bella
2	Luna

Comanda

PK_NrComanda	FK_IdClient	Cantitate
10	1	9
11	2	50

Factura

FK_NrComanda	FK_IdClient	Data
10	1	18/12/2021
11	2	23/12/2021

6. Originalitate

