Proiect - Baze de date

Nume: Ciutacu Claudia

Grupa: 1050 C

**Cuprins**

1. Tema Proiectului
2. Descrierea succintă a proiectului
3. Structura tabelelor

1.descriere

2.tipuri de date

3.relaţii dintre tabele

1. Schema conceptuală a bazei de date
2. Popularea tabelelor
3. Exerciții proiect

**I. Tema proiectului**

Am ales să proiectez și să implementez o bază de date pentru gesiunea activității unui supermarket.

**II.Descrierea succintă a proiectului**

Baza de date creată de mine reprezintă un mijloc prin care managerul unui supermarket poate gestiona detalii legate de angajați, furnizori, produsele puse la vânzare, cât și comenzile plasate de clienți.

În mare parte, angajații sunt caracterizați de funcția pe care o dețin și comenzile de care se ocupă fiecare. La rândul lor, comenzile sunt caracterizate de produsele ce intră in componența lor, fiecare produs având un anumit furnizor.

**III.Structura tabelelor**

* Angajati\_proiect: tabelă ce conține datele personale ale angajaților, cât și id-ul, salariuș si codurile asociate funcției și superiorului.Fiecare angajat ocupă o singură funcție, lucrează la un singur sediu și se poate ocupa de mai multe comenzi.

|  |  |  |
| --- | --- | --- |
| Nume coloană | Tip restricție | Tip de date |
| ID\_ANGAJAT | Primary key | Number(3) |
| NUME | - | Varchar2(20) |
| PRENUME | - | Varchar2(20) |
| ADRESĂ | - | Varchar2(50) |
| TELEFON | - | Varchar2(10) |
| EMAIL | - | Varchar2(20) |
| COD\_FUNCȚIE | Foreign key | Number(3) |
| NUME\_ORAȘ | Foreign key | Varchar2(20) |
| DATA\_ANGAJĂRII | - | Date |
| COMISION | - | Number(2,2) |
| SALARIU | - | Number(8,2) |
| COD\_SUPERIOR | - | Number(4) |

* Funcție\_proiect: tabela conține detaliile funcțiilor la nivelul supermarketului. O funcție poate fi ocupată de mai mulți angajați.

|  |  |  |
| --- | --- | --- |
| Nume coloană | Tip restricție | Tip de date |
| COD\_FUNCȚIE | Primary key | Number(3) |
| NUME\_FUNCȚIE | - | Varchar2(30) |
| ATRIBUȚII | - | Varchar2(50) |
| NR\_ANGAJAȚI\_PE\_FUNCȚIE | - | Number(2) |

* Sediu\_proiect: într-un sediu pot exista mai mulți angajați

|  |  |  |
| --- | --- | --- |
| Nume coloană | Tip restricție | Tip de date |
| NUME\_ORAȘ | Primary key | Varchar2(20) |
| ADRESĂ | - | Varchar2(30) |

* Comenzi\_proiect: un angajat se poate ocupa de mai multe comenzi, o comandă poate avea mai multe produse.

|  |  |  |
| --- | --- | --- |
| Nume coloană | Tip restricție | Tip de date |
| COD\_COMANDĂ | Primary key | Number(5) |
| NUME\_PRODUSE | - | Varchar2(70) |
| PREȚ\_TOTAL | - | Number(5) |
| ID\_ANGAJAT | Foreign key | Number(3) |

* Produse\_proiect: un produs aparține unei singure comenzi și unui singur furnizor.

|  |  |  |
| --- | --- | --- |
| Nume coloană | Tip restricție | Tip de date |
| COD\_ARTICOL | Primary key | Number(5) |
| DENUMIRE | - | Varchar2(20) |
| PREȚ | - | Number(5) |
| COD\_COMANDĂ | Foreign key | Number(5) |
| COD\_FURNIZOR | Foreign key | Number(5) |
| CANTITATE | - | Number(3) |

* Furnizor\_proiect: un furnizor poate aproviziona supermarketul cu mai multe produse.

|  |  |  |
| --- | --- | --- |
| Nume coloană | Tip restricție | Tip de date |
| COD\_FURNIZOR | Primary key | Number(5) |
| NUME\_FURNIZOR | - | Varchar2(20) |
| ADRESĂ\_DEPOZIT | - | Varchar2(50) |
| TELEFON\_FURNIZOR | - | Number(10) |
| EMAIL\_FURNIZOR | - | Varchar2(50) |

**IV.Schema conceptuală a bazei de date**

Diagram

Description automatically generated

|  |  |  |
| --- | --- | --- |
| **CREARE TABELA ANGAJATI**  CREATE TABLE ANGAJATI\_PROIECT(  ID\_ANGAJAT NUMBER(3),  NUME VARCHAR(20) ,  PRENUME VARCHAR(20),  ADRESA VARCHAR(50),  TELEFON VARCHAR(10),  EMAIL VARCHAR(20)  ); | **CREARE TABELA PRODUSE**  CREATE TABLE PRODUSE\_PROIECT(  COD\_ARTICOL NUMBER(5),  DENUMIRE VARCHAR(20),  MATERIAL VARCHAR(20),  PRET NUMBER(5)  ); | **CREARE TABELA FUNCTIE**  CREATE TABLE FUNCTIE\_PROIECT(  COD\_FUNCTIE NUMBER(3),  NUME\_FUNCTIE VARCHAR(30),  ATRIBUTII VARCHAR(50),  NR\_ANGAJATI\_PE\_FUNCTIE NUMBER(2)  ); |
| **CREARE TABELA FURNIZOR**  CREATE TABLE FURNIZOR\_PROIECT(  COD\_FURNIZOR NUMBER(5),  NUME\_FURNIZOR VARCHAR(20),  ADRESA\_DEPOZIT VARCHAR(20),  TELEFON\_FURNIZOR NUMBER(10),  EMAIL\_FURNIZOR VARCHAR(20)  ); | **CREARE TABELA COMENZI**  CREATE TABLE COMENZI\_PROIECT(  COD\_COMANDA NUMBER(5) PRIMARY KEY,  NUME\_PRODUSE VARCHAR(20),  PRET\_TOTAL NUMBER(5)  ); | **CREARE TABELA SEDIU**  CREATE TABLE SEDIU\_PROIECT(  NUME\_ORAS VARCHAR(20) PRIMARY KEY,  ADRESA VARCHAR(30)  ); |

**V.Popularea tabelelor**

**Crearea tabelelor**

**Restricțiile de integritate**

|  |  |
| --- | --- |
| **RESTRICTII : CHEILE PRIMARE**  ALTER TABLE ANGAJATI\_PROIECT  ADD CONSTRAINT ID\_ANGAJAT\_PK primary key (ID\_ANGAJAT);  ALTER TABLE PRODUSE\_PROIECT  ADD CONSTRAINT COD\_ARTICOL\_PK primary key (COD\_ARTICOL);  ALTER TABLE SEDIU\_PROIECT  ADD CONSTRAINT SEDIU\_PROIECT\_PK primary key (NUME\_ORAS);  ALTER TABLE FUNCTIE\_PROIECT  ADD CONSTRAINT FUNCTIE\_PROIECT\_PK primary key (COD\_FUNCTIE);  ALTER TABLE COMENZI\_PROIECT  ADD CONSTRAINT COMENZI\_PROIECT\_PK primary key (COD\_COMANDA);  ALTER TABLE FURNIZOR\_PROIECT  ADD CONSTRAINT FURNIZOR\_PROIECT\_PK primary key (COD\_FURNIZOR); | **RESTRICTII: CHEILE EXTERNE**  ALTER TABLE ANGAJATI\_PROIECT  ADD CONSTRAINT ANGAJATI\_PROIECT\_FK foreign key (COD\_FUNCTIE)  REFERENCES FUNCTIE\_PROIECT(COD\_FUNCTIE);  ALTER TABLE ANGAJATI\_PROIECT  ADD CONSTRAINT SEDIU\_FK foreign key (NUME\_ORAS)  REFERENCES SEDIU\_PROIECT(NUME\_ORAS);  ALTER TABLE COMENZI\_PROIECT  ADD CONSTRAINT ID\_ANGAJAT\_FK foreign key (ID\_ANGAJAT)  REFERENCES ANGAJATI\_PROIECT(ID\_ANGAJAT);  ALTER TABLE PRODUSE\_PROIECT  ADD CONSTRAINT COMANDA\_FK foreign key (COD\_COMANDA)  REFERENCES COMENZI\_PROIECT(COD\_COMANDA);  ALTER TABLE PRODUSE\_PROIECT  ADD CONSTRAINT FURNIZOR\_FK foreign key (COD\_FURNIZOR)  REFERENCES FURNIZOR\_PROIECT(COD\_FURNIZOR); |

* **POPULAREA TABELEI FURNIZOR**

INSERT INTO FURNIZOR\_PROIECT(COD\_FURNIZOR,NUME\_FURNIZOR,ADRESA\_DEPOZIT,TELEFON\_FURNIZOR,EMAIL\_FURNIZOR)

VALUES(1000,'Belder SRL','Str.Zorilor,nr.10,Bucuresti','0765423654','Beldersrl@yahoo.com');

INSERT INTO FURNIZOR\_PROIECT(COD\_FURNIZOR,NUME\_FURNIZOR,ADRESA\_DEPOZIT,TELEFON\_FURNIZOR,EMAIL\_FURNIZOR)

VALUES(1001,'Fonzi Tranding','Str.Narciselor,nr.87,Bacau','0763425743','fonzitranding@yahoo.com');

INSERT INTO FURNIZOR\_PROIECT(COD\_FURNIZOR,NUME\_FURNIZOR,ADRESA\_DEPOZIT,TELEFON\_FURNIZOR,EMAIL\_FURNIZOR)

VALUES(1002,'Frigmecanic SRL','Str.iasomiilor,nr.76,Ploiesti','0734526789','frigmecanic@gmail.com');

INSERT INTO FURNIZOR\_PROIECT(COD\_FURNIZOR,NUME\_FURNIZOR,ADRESA\_DEPOZIT,TELEFON\_FURNIZOR,EMAIL\_FURNIZOR)

VALUES(1003,'Sam Bros','Str.Garii,nr.23,Timisoara','0724356871','sambros@yahoo.com');

INSERT INTO FURNIZOR\_PROIECT(COD\_FURNIZOR,NUME\_FURNIZOR,ADRESA\_DEPOZIT,TELEFON\_FURNIZOR,EMAIL\_FURNIZOR)

VALUES(1004,'Maicom SRL','Stefan Cel Mare,nr.12,Iasi','0765423654','maicomsrl@yahoo.com');

INSERT INTO FURNIZOR\_PROIECT(COD\_FURNIZOR,NUME\_FURNIZOR,ADRESA\_DEPOZIT,TELEFON\_FURNIZOR,EMAIL\_FURNIZOR)

VALUES(1005,'Lidarin SRL','Str.Viilor,nr.45,Suceava','0765423244','lidarin@gmail.com');

INSERT INTO FURNIZOR\_PROIECT(COD\_FURNIZOR,NUME\_FURNIZOR,ADRESA\_DEPOZIT,TELEFON\_FURNIZOR,EMAIL\_FURNIZOR)

VALUES(1006,'Magister SRL','Str.Independentei,nr.29,Deva','0798456321','magister@yahoo.com');

INSERT INTO FURNIZOR\_PROIECT(COD\_FURNIZOR,NUME\_FURNIZOR,ADRESA\_DEPOZIT,TELEFON\_FURNIZOR,EMAIL\_FURNIZOR)

VALUES(1007,'Expert SRL','Str.Campului,nr.37,Mehedinti','0742687529','expertsrl@gmail.com');

INSERT INTO FURNIZOR\_PROIECT(COD\_FURNIZOR,NUME\_FURNIZOR,ADRESA\_DEPOZIT,TELEFON\_FURNIZOR,EMAIL\_FURNIZOR)

VALUES(1008,'Dincov SRL','Str.Stadionului,nr.11,Brasov','0725843189','dincovsrl@gmail.com');

INSERT INTO FURNIZOR\_PROIECT(COD\_FURNIZOR,NUME\_FURNIZOR,ADRESA\_DEPOZIT,TELEFON\_FURNIZOR,EMAIL\_FURNIZOR)

VALUES(1009,'Darolex SRL','Str.Unirii,nr.23,Predeal','0789423561','darolex@gmail.com');

* **POPULAREA TABELEI FUNCTIE**

INSERT INTO FUNCTIE\_PROIECT(COD\_FUNCTIE,NUME\_FUNCTIE,ATRIBUTII,NR\_ANGAJATI\_PE\_FUNCTIE)

VALUES(100,'Casier','Scanare produse',10);

INSERT INTO FUNCTIE\_PROIECT(COD\_FUNCTIE,NUME\_FUNCTIE,ATRIBUTII,NR\_ANGAJATI\_PE\_FUNCTIE)

VALUES(101,'Manager','Implementare proiect de afaceri',1);

INSERT INTO FUNCTIE\_PROIECT(COD\_FUNCTIE,NUME\_FUNCTIE,ATRIBUTII,NR\_ANGAJATI\_PE\_FUNCTIE)

VALUES(102,'Agent de paza','Paza si siguranta clientilor',3);

INSERT INTO FUNCTIE\_PROIECT(COD\_FUNCTIE,NUME\_FUNCTIE,ATRIBUTII,NR\_ANGAJATI\_PE\_FUNCTIE)

VALUES(103,'Femeie de serviciu','Curatenie',4);

INSERT INTO FUNCTIE\_PROIECT(COD\_FUNCTIE,NUME\_FUNCTIE,ATRIBUTII,NR\_ANGAJATI\_PE\_FUNCTIE)

VALUES(104,'Contabil','Facturi si chitante',2);

INSERT INTO FUNCTIE\_PROIECT(COD\_FUNCTIE,NUME\_FUNCTIE,ATRIBUTII,NR\_ANGAJATI\_PE\_FUNCTIE)

VALUES(105,'Sef depozit','Gestionarea produselor din depozit',1);

INSERT INTO FUNCTIE\_PROIECT(COD\_FUNCTIE,NUME\_FUNCTIE,ATRIBUTII,NR\_ANGAJATI\_PE\_FUNCTIE)

VALUES(106,'Sef raion','Gestionarea produselor de pe raft',2);

INSERT INTO FUNCTIE\_PROIECT(COD\_FUNCTIE,NUME\_FUNCTIE,ATRIBUTII,NR\_ANGAJATI\_PE\_FUNCTIE)

VALUES(107,'Vanzator','Vanzarea diferitelor produse',8);

INSERT INTO FUNCTIE\_PROIECT(COD\_FUNCTIE,NUME\_FUNCTIE,ATRIBUTII,NR\_ANGAJATI\_PE\_FUNCTIE)

VALUES(108,'Sef tura','Coordonarea activitatii personalului',1);

INSERT INTO FUNCTIE\_PROIECT(COD\_FUNCTIE,NUME\_FUNCTIE,ATRIBUTII,NR\_ANGAJATI\_PE\_FUNCTIE)

VALUES(109,'Gestionar magazin','Receptia Calitativa si cantitativa a marfurilor',1);

* **POPULAREA TABELEI SEDIU**

INSERT INTO SEDIU\_PROIECT(NUME\_ORAS,ADRESA)

VALUES('Busteni','Str.Ranca,nr.10');

INSERT INTO SEDIU\_PROIECT(NUME\_ORAS,ADRESA)

VALUES('Brasov','Str.Zambilelor,nr.32');

INSERT INTO SEDIU\_PROIECT(NUME\_ORAS,ADRESA)

VALUES('Bucuresti','Str.Mihai Viteazu,nr.76');

INSERT INTO SEDIU\_PROIECT(NUME\_ORAS,ADRESA)

VALUES('Cluj-Napoca','Str.Maceselor,nr.83');

INSERT INTO SEDIU\_PROIECT(NUME\_ORAS,ADRESA)

VALUES('Ploiesti','Str.Josnica,nr.25');

INSERT INTO SEDIU\_PROIECT(NUME\_ORAS,ADRESA)

VALUES('Sibiu','Str.Liliacului,nr.40');

INSERT INTO SEDIU\_PROIECT(NUME\_ORAS,ADRESA)

VALUES('Timisoara','Str.Mihai Bravu,nr.69');

INSERT INTO SEDIU\_PROIECT(NUME\_ORAS,ADRESA)

VALUES('Iasi','Str.Plopilor,nr.14');

INSERT INTO SEDIU\_PROIECT(NUME\_ORAS,ADRESA)

VALUES('Oradea','Str.Unirii,nr.64');

INSERT INTO SEDIU\_PROIECT(NUME\_ORAS,ADRESA)

VALUES('Constanta','Str.Republicii,nr.23');

* **POPULAREA TABELEI ANGAJATI**

INSERT INTO ANGAJATI\_PROIECT(ID\_ANGAJAT,NUME,PRENUME,ADRESA,TELEFON,EMAIL,COD\_FUNCTIE,NUME\_ORAS)

VALUES(1,'Ciutacu','Claudia','Str.Parcului,nr.9','0730214155','claudiaciutacu@yahoo.com',100,'Busteni');

INSERT INTO ANGAJATI\_PROIECT(ID\_ANGAJAT,NUME,PRENUME,ADRESA,TELEFON,EMAIL,COD\_FUNCTIE,NUME\_ORAS)

VALUES(2,'Coman','Emilia','Str.Crudului,nr.45','0732474697','comanemilia@gmail.com',102,'Ploiesti');

INSERT INTO ANGAJATI\_PROIECT(ID\_ANGAJAT,NUME,PRENUME,ADRESA,TELEFON,EMAIL,COD\_FUNCTIE,NUME\_ORAS)

VALUES(3,'Daia','Luca','Str.Sperantei,nr.10','0724561972','daialuca@yahoo.com',101,'Brasov');

INSERT INTO ANGAJATI\_PROIECT(ID\_ANGAJAT,NUME,PRENUME,ADRESA,TELEFON,EMAIL,COD\_FUNCTIE,NUME\_ORAS)

VALUES(4,'Craciun','Catalin','Str.Independentei,nr.38','0705127845','craciuncatalin@yahoo.com',104,'Cluj-Napoca');

INSERT INTO ANGAJATI\_PROIECT(ID\_ANGAJAT,NUME,PRENUME,ADRESA,TELEFON,EMAIL,COD\_FUNCTIE,NUME\_ORAS)

VALUES(5,'Danila','Daniel','Str.Teilor,nr.41','0767561245','daniladaniel@gmail.com',105,'Iasi');

INSERT INTO ANGAJATI\_PROIECT(ID\_ANGAJAT,NUME,PRENUME,ADRESA,TELEFON,EMAIL,COD\_FUNCTIE,NUME\_ORAS)

VALUES(6,'Cozma','Ciprian','Str.Closca,nr.99','0735547821','cozmaciprian@yahoo.com',106,'Timisoara');

INSERT INTO ANGAJATI\_PROIECT(ID\_ANGAJAT,NUME,PRENUME,ADRESA,TELEFON,EMAIL,COD\_FUNCTIE,NUME\_ORAS)

VALUES(7,'Cotoi','Andrei','Str.Traian,nr.22','0712567832','cotoiandrei@yahoo.com',107,'Bucuresti');

INSERT INTO ANGAJATI\_PROIECT(ID\_ANGAJAT,NUME,PRENUME,ADRESA,TELEFON,EMAIL,COD\_FUNCTIE,NUME\_ORAS)

VALUES(8,'Cirstea','Andrei','Str.Rozelor,nr.21','0774310964','cirsteaandrei@gmail.com',108,'Oradea');

INSERT INTO ANGAJATI\_PROIECT(ID\_ANGAJAT,NUME,PRENUME,ADRESA,TELEFON,EMAIL,COD\_FUNCTIE,NUME\_ORAS)

VALUES(9,'Stoica','Valentin','Str.Albului,nr.12','0766718921','stoicavalentin@gmail.com',109,'Constanta');

INSERT INTO ANGAJATI\_PROIECT(ID\_ANGAJAT,NUME,PRENUME,ADRESA,TELEFON,EMAIL,COD\_FUNCTIE,NUME\_ORAS)

VALUES(10,'Ion','Vasilica','Str. 1 mai,nr.10','0767129021','ionvasilica@yahoo.com',103,'Sibiu');

* **POPULAREA TABELEI COMENZI**

INSERT INTO COMENZI\_PROIECT(COD\_COMANDA,NUME\_PRODUSE,PRET\_TOTAL,ID\_ANGAJAT)

VALUES(234,'cartofi,paine,suc',34,7);

INSERT INTO COMENZI\_PROIECT(COD\_COMANDA,NUME\_PRODUSE,PRET\_TOTAL,ID\_ANGAJAT)

VALUES(235,'bere,vin,ciocolata,matura',124,5);

INSERT INTO COMENZI\_PROIECT(COD\_COMANDA,NUME\_PRODUSE,PRET\_TOTAL,ID\_ANGAJAT)

VALUES(224,'banane',9,5);

INSERT INTO COMENZI\_PROIECT(COD\_COMANDA,NUME\_PRODUSE,PRET\_TOTAL,ID\_ANGAJAT)

VALUES(112,'sampon,sapun,rosii,apa,salam',67,8);

INSERT INTO COMENZI\_PROIECT(COD\_COMANDA,NUME\_PRODUSE,PRET\_TOTAL,ID\_ANGAJAT)

VALUES(565,'cartofi,salata,patrunjel,lavete,cos de gunoi',90,6);

INSERT INTO COMENZI\_PROIECT(COD\_COMANDA,NUME\_PRODUSE,PRET\_TOTAL,ID\_ANGAJAT)

VALUES(888,'pateu,salam,iaurt,paine,vodka',145,6);

INSERT INTO COMENZI\_PROIECT(COD\_COMANDA,NUME\_PRODUSE,PRET\_TOTAL,ID\_ANGAJAT)

VALUES(254,'branza,cascaval,paste,sos de rosii,sare,busuioc',91,5);

INSERT INTO COMENZI\_PROIECT(COD\_COMANDA,NUME\_PRODUSE,PRET\_TOTAL,ID\_ANGAJAT)

VALUES(444,'ketchup,masline,minge,bicicleta',1243,7);

INSERT INTO COMENZI\_PROIECT(COD\_COMANDA,NUME\_PRODUSE,PRET\_TOTAL,ID\_ANGAJAT)

VALUES(954,'apa,bere,sushi',71,9);

INSERT INTO COMENZI\_PROIECT(COD\_COMANDA,NUME\_PRODUSE,PRET\_TOTAL,ID\_ANGAJAT)

VALUES(116,'televizor,cuptor cu microunde',12365,9);

INSERT INTO COMENZI\_PROIECT(COD\_COMANDA,NUME\_PRODUSE,PRET\_TOTAL,ID\_ANGAJAT)

VALUES(298,'bere,prajituri,stafide',69,5);

* **POPULAREA TABELEI PRODUSE**

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(12,'cartofi',3,234,1000);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(13,'rosii',5,112,1000);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(45,'banane',4,224,1008);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(89,'paine',2,888,1001);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(33,'vin',70,235,1002);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(22,'salata',2,565,1000);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(32,'ketchup',8,444,1005);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(47,'sampon',23,112,1006);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(17,'apa',2,954,1007);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(18,'televizor',12129,116,1004);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(24,'cuptor cu microunde',236,116,1004);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(71,'suc',29,234,1007);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(90,'bere',6,235,1007);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(21,'ciocolata',11,235,1001);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(55,'matura',37,235,1001);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(20,'sapun',9,112,1006);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(23,'salam',28,112,1003);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(25,'patrunjel',1,565,1000);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(26,'lavete',11,565,1008);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(27,'cos de gunoi',73,565,1008);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(28,'pateu',5,888,1005);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(29,'iaurt',13,888,1001);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(31,'vodka',97,888,1007);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(34,'branza',19,254,1003);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(36,'cascaval',23,254,1003);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(37,'paste',5,254,1000);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(50,'sos de rosii',16,254,1001);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(51,'sare',4,254,1001);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(52,'busuioc',24,254,1000);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(61,'masline',20,444,1009);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(62,'minge',56,888,1008);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(63,'bicicleta',1159,444,1008);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(64,'sushi',63,954,1004);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(65,'paine',2,234,1001);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(66,'apa',2,112,1001);

INSERT INTO PRODUSE\_PROIECT(COD\_ARTICOL,DENUMIRE,PRET,COD\_COMANDA,COD\_FURNIZOR)

VALUES(67,'bere',6,298,1001);

* **UPDATE-URI**

UPDATE ANGAJATI\_PROIECT SET NUME='Cristea' WHERE ID\_ANGAJAT=8;

ALTER TABLE ANGAJATI\_PROIECT

ADD COD\_FUNCTIE NUMBER(3);

ALTER TABLE ANGAJATI\_PROIECT

ADD NUME\_ORAS VARCHAR(20);

ALTER TABLE COMENZI\_PROIECT

ADD ID\_ANGAJAT NUMBER(3);

ALTER TABLE PRODUSE\_PROIECT

ADD COD\_COMANDA NUMBER(5);

ALTER TABLE PRODUSE\_PROIECT

ADD COD\_FURNIZOR NUMBER(5);

ALTER TABLE PRODUSE\_PROIECT DROP COLUMN MATERIAL;

ALTER TABLE COMENZI\_PROIECT MODIFY NUME\_PRODUSE VARCHAR(70);

ALTER TABLE FUNCTIE\_PROIECT MODIFY ATRIBUTII VARCHAR(50);

ALTER TABLE ANGAJATI\_PROIECT MODIFY EMAIL VARCHAR(30);

ALTER TABLE FURNIZOR\_PROIECT MODIFY EMAIL\_FURNIZOR VARCHAR(50);

ALTER TABLE FURNIZOR\_PROIECT MODIFY ADRESA\_DEPOZIT VARCHAR(50);

DELETE FROM FURNIZOR\_PROIECT WHERE COD\_FURNIZOR=1000;

ALTER TABLE ANGAJATI\_PROIECT

ADD COD\_MANAGER\_SUPERIOR VARCHAR(3);

ALTER TABLE ANGAJATI\_PROIECT

DROP COLUMN COD\_MANAGER\_SUPERIOR;

ALTER TABLE ANGAJATI\_PROIECT

ADD COD\_SUPERIOR NUMBER(4);

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=104 WHERE COD\_FUNCTIE=100;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=NULL WHERE COD\_FUNCTIE=101;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=108 WHERE COD\_FUNCTIE=102;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=108 WHERE COD\_FUNCTIE=103;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=101 WHERE COD\_FUNCTIE=104;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=108 WHERE COD\_FUNCTIE=105;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=108 WHERE COD\_FUNCTIE=106;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=100 WHERE COD\_FUNCTIE=107;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=101 WHERE COD\_FUNCTIE=108;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=101 WHERE COD\_FUNCTIE=109;

INSERT INTO ANGAJATI\_PROIECT(ID\_ANGAJAT,NUME,PRENUME,ADRESA,TELEFON,EMAIL,COD\_FUNCTIE,NUME\_ORAS)

VALUES(11,'Stoica','Bianca','Str.Amarului,nr.12','0766710921','stoicabianca@gmail.com',106,'Constanta');

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=106 WHERE ID\_ANGAJAT=11;

ALTER TABLE ANGAJATI\_PROIECT ADD DATA\_ANGAJARII DATE;

UPDATE ANGAJATI\_PROIECT SET DATA\_ANGAJARII=to\_date('11-12-2015','dd-mm-yyyy') WHERE ID\_ANGAJAT=1;

UPDATE ANGAJATI\_PROIECT SET DATA\_ANGAJARII=to\_date('4-11-2015','dd-mm-yyyy') WHERE ID\_ANGAJAT=2;

UPDATE ANGAJATI\_PROIECT SET DATA\_ANGAJARII=to\_date('21-01-2015','dd-mm-yyyy') WHERE ID\_ANGAJAT=3;

UPDATE ANGAJATI\_PROIECT SET DATA\_ANGAJARII=to\_date('15-10-2021','dd-mm-yyyy') WHERE ID\_ANGAJAT=4;

UPDATE ANGAJATI\_PROIECT SET DATA\_ANGAJARII=to\_date('17-10-2009','dd-mm-yyyy') WHERE ID\_ANGAJAT=5;

UPDATE ANGAJATI\_PROIECT SET DATA\_ANGAJARII=to\_date('17-12-2013','dd-mm-yyyy') WHERE ID\_ANGAJAT=6;

UPDATE ANGAJATI\_PROIECT SET DATA\_ANGAJARII=to\_date('16-10-2021','dd-mm-yyyy') WHERE ID\_ANGAJAT=7;

UPDATE ANGAJATI\_PROIECT SET DATA\_ANGAJARII=to\_date('31-10-2020','dd-mm-yyyy') WHERE ID\_ANGAJAT=8;

UPDATE ANGAJATI\_PROIECT SET DATA\_ANGAJARII=to\_date('23-06-2019','dd-mm-yyyy') WHERE ID\_ANGAJAT=9;

UPDATE ANGAJATI\_PROIECT SET DATA\_ANGAJARII=to\_date('10-11-2018','dd-mm-yyyy') WHERE ID\_ANGAJAT=10;

UPDATE ANGAJATI\_PROIECT SET DATA\_ANGAJARII=to\_date('27-11-2007','dd-mm-yyyy') WHERE ID\_ANGAJAT=11;

ALTER TABLE ANGAJATI\_PROIECT ADD SALARIU NUMBER(5,2);

ALTER TABLE ANGAJATI\_PROIECT ADD COMISION NUMBER(2,2);

ALTER TABLE ANGAJATI\_PROIECT MODIFY SALARIU NUMBER(8,2);

UPDATE ANGAJATI\_PROIECT SET SALARIU=2340 WHERE ID\_ANGAJAT=1;

UPDATE ANGAJATI\_PROIECT SET SALARIU=3240 WHERE ID\_ANGAJAT=2;

UPDATE ANGAJATI\_PROIECT SET SALARIU=5400 WHERE ID\_ANGAJAT=3;

UPDATE ANGAJATI\_PROIECT SET SALARIU=2340 WHERE ID\_ANGAJAT=4;

UPDATE ANGAJATI\_PROIECT SET SALARIU=2234 WHERE ID\_ANGAJAT=5;

UPDATE ANGAJATI\_PROIECT SET SALARIU=1678 WHERE ID\_ANGAJAT=6;

UPDATE ANGAJATI\_PROIECT SET SALARIU=1908 WHERE ID\_ANGAJAT=7;

UPDATE ANGAJATI\_PROIECT SET SALARIU=1234 WHERE ID\_ANGAJAT=8;

UPDATE ANGAJATI\_PROIECT SET SALARIU=3214 WHERE ID\_ANGAJAT=9;

UPDATE ANGAJATI\_PROIECT SET SALARIU=2132 WHERE ID\_ANGAJAT=10;

UPDATE ANGAJATI\_PROIECT SET SALARIU=3456 WHERE ID\_ANGAJAT=11;

UPDATE ANGAJATI\_PROIECT SET COMISION=null;

UPDATE ANGAJATI\_PROIECT SET COMISION=null WHERE ID\_ANGAJAT=3;

UPDATE ANGAJATI\_PROIECT SET COMISION=0.1 WHERE ID\_ANGAJAT=8;

UPDATE ANGAJATI\_PROIECT SET COMISION=0.12 WHERE ID\_ANGAJAT=5;

ALTER TABLE ANGAJATI\_PROIECT

DROP COLUMN COD\_MANAGER\_SUPERIOR;

ALTER TABLE ANGAJATI\_PROIECT

ADD COD\_SUPERIOR VARCHAR(3);

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=104 WHERE COD\_FUNCTIE=100;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=NULL WHERE COD\_FUNCTIE=101;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=108 WHERE COD\_FUNCTIE=102;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=108 WHERE COD\_FUNCTIE=103;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=101 WHERE COD\_FUNCTIE=104;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=108 WHERE COD\_FUNCTIE=105;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=108 WHERE COD\_FUNCTIE=106;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=100 WHERE COD\_FUNCTIE=107;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=101 WHERE COD\_FUNCTIE=108;

UPDATE ANGAJATI\_PROIECT SET COD\_SUPERIOR=101 WHERE COD\_FUNCTIE=109;

ALTER TABLE PRODUSE\_PROIECT

ADD CANTITATE NUMBER(3);

UPDATE PRODUSE\_PROIECT SET CANTITATE = 3

WHERE COD\_ARTICOL=45;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 4

WHERE COD\_ARTICOL=89;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 28

WHERE COD\_ARTICOL=33;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 51

WHERE COD\_ARTICOL=32;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 10

WHERE COD\_ARTICOL=47;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 23

WHERE COD\_ARTICOL=17;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 2

WHERE COD\_ARTICOL=18;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 6

WHERE COD\_ARTICOL=24;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 43

WHERE COD\_ARTICOL=71;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 32

WHERE COD\_ARTICOL=90;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 12

WHERE COD\_ARTICOL=21;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 2

WHERE COD\_ARTICOL=55;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 7

WHERE COD\_ARTICOL=20;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 9

WHERE COD\_ARTICOL=23;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 12

WHERE COD\_ARTICOL=26;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 3

WHERE COD\_ARTICOL=27;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 12

WHERE COD\_ARTICOL=28;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 11

WHERE COD\_ARTICOL=29;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 8

WHERE COD\_ARTICOL=31;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 6

WHERE COD\_ARTICOL=34;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 4

WHERE COD\_ARTICOL=36;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 3

WHERE COD\_ARTICOL=50;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 10

WHERE COD\_ARTICOL=51;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 20

WHERE COD\_ARTICOL=61;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 41

WHERE COD\_ARTICOL=62;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 3

WHERE COD\_ARTICOL=63;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 12

WHERE COD\_ARTICOL=64;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 12

WHERE COD\_ARTICOL=65;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 30

WHERE COD\_ARTICOL=66;

UPDATE PRODUSE\_PROIECT SET CANTITATE = 7

WHERE COD\_ARTICOL=67;

**VI.Exerciții proiect**

* **INTEROGAREA TABELELOR**

1.Sa se selecteze toti angajatii din tabela ANGAJATI\_PROIECT.

SELECT \* FROM ANGAJATI\_PROIECT;

Graphical user interface, text, application

Description automatically generated

2.Sa se selecteze coloanele id\_angajat, nume, prenume si cod\_functie din tabela ANGAJATI\_PROIECT.

SELECT ID\_ANGAJAT,NUME,PRENUME,COD\_FUNCTIE FROM ANGAJATI\_PROIECT;

Table

Description automatically generated

3.Sa se selecteze doar angajatii al caror email contin 'gmail'.

SELECT \* FROM ANGAJATI\_PROIECT

WHERE lower(email) LIKE '%gmail%';

Graphical user interface, application

Description automatically generated

4.Sa se selecteze comenzile incheiate de salariatul cu id\_angajat = 6.

SELECT \* FROM COMENZI\_PROIECT

WHERE ID\_ANGAJAT=6

ORDER BY COD\_COMANDA;

Graphical user interface, text, application

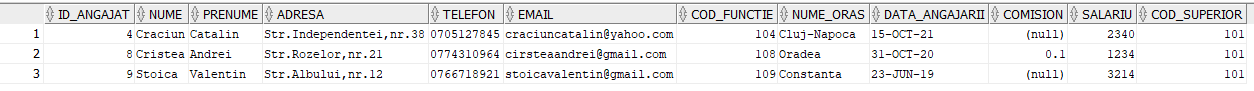
Description automatically generated

5.Sa se selecteze toti angajatii care au fost angajati dupa 1 ianuarie 2019 si care au cod\_superior=101.

SELECT \* FROM ANGAJATI\_PROIECT

WHERE COD\_SUPERIOR=101

AND DATA\_ANGAJARII>TO\_DATE('01.01.2019','DD.MM.YYYY');



6.Sa se selecteze id\_angajat, nume, prenume, cod\_functie, din tabela angajati\_proiect si nume\_functie din tabela departamente si sa se realizeze jonctiunea dintre cele doua tabele

SELECT a.ID\_ANGAJAT, a.NUME, a.PRENUME, a.COD\_FUNCTIE, f.nume\_functie

FROM ANGAJATI\_PROIECT a, FUNCTIE\_PROIECT f

WHERE a.COD\_FUNCTIE=f.COD\_FUNCTIE;

Table

Description automatically generated

7.Sa se afiseze cod\_articol,denumire pentru produsele care nu sunt livrate de furnizor cu cod\_furnizor=1007 si al caror pret este mai mare decat oricare dintre preturile produselor livrate de furnizorul cu codul 1007.

SELECT COD\_ARTICOL,DENUMIRE,PRET,COD\_FURNIZOR

FROM PRODUSE\_PROIECT

WHERE PRET>ANY

(SELECT PRET FROM PRODUSE\_PROIECT WHERE COD\_FURNIZOR=1007)

AND COD\_FURNIZOR<>1007

ORDER BY PRET DESC;

Table

Description automatically generatedTable

Description automatically generated with medium confidence

8.Sa se afiseze cod\_articol,denumire pentru produsele care nu sunt livrate de furnizor cu cod\_furnizor=1007 si al caror pret este mai mare decat fiecare dintre preturile produselor livrate de furnizorul cu codul 1007.

SELECT COD\_ARTICOL,DENUMIRE,PRET,COD\_FURNIZOR

FROM PRODUSE\_PROIECT

WHERE PRET>ALL

(SELECT PRET FROM PRODUSE\_PROIECT WHERE COD\_FURNIZOR=1007)

AND COD\_FURNIZOR<>1007

ORDER BY PRET DESC;

Graphical user interface, text, application

Description automatically generatedTable

Description automatically generated with medium confidence

9.Sa se selecteze comenzile incheiate de angajati .

SELECT a.\*, c.\*

FROM ANGAJATI\_PROIECT a, COMENZI\_PROIECT c

WHERE a.ID\_ANGAJAT= c.ID\_ANGAJAT; Graphical user interface, application

Description automatically generatedA picture containing chart

Description automatically generated

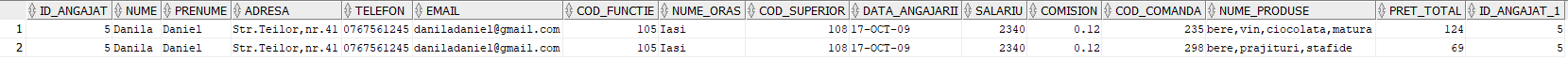
10.Sa se selecteze comenzile incheiate de angajatul Danila Daniel care contin produsul bere.

SELECT a.\*, c.\*

FROM ANGAJATI\_PROIECT a, COMENZI\_PROIECT c

WHERE a.ID\_ANGAJAT=c.ID\_ANGAJAT

AND lower (c.NUME\_PRODUSE) LIKE '%bere%'

AND (a.NUME)='Danila'; 

11.Sa se calculeze valoarea fiecarui produs (valoare = cantitate \* pret) si sa se afiseze furnizorul produsului, denumirea produsului, pretul, cantitatea si valoarea.

SELECT FURNIZOR\_PROIECT.NUME\_FURNIZOR,

PRODUSE\_PROIECT.DENUMIRE,PRODUSE\_PROIECT.CANTITATE,PRODUSE\_PROIECT.PRET,PRODUSE\_PRO

IECT.CANTITATE\*PRODUSE\_PROIECT.PRET AS VALOARE

FROM FURNIZOR\_PROIECT,PRODUSE\_PROIECT

WHERE FURNIZOR\_PROIECT.COD\_FURNIZOR=PRODUSE\_PROIECT.COD\_FURNIZOR;

Graphical user interface, table

Description automatically generated

Table

Description automatically generated

12.Sa se selecteze numai produsele cu valoarea cuprinsa intre 100 si 700 alaturi de furnizorul lor.

SELECT FURNIZOR\_PROIECT.NUME\_FURNIZOR,

PRODUSE\_PROIECT.DENUMIRE,PRODUSE\_PROIECT.CANTITATE,PRODUSE\_PROIECT.PRET,PRODUSE\_PRO

IECT.CANTITATE\*PRODUSE\_PROIECT.PRET AS VALOARE

FROM FURNIZOR\_PROIECT,PRODUSE\_PROIECT

WHERE FURNIZOR\_PROIECT.COD\_FURNIZOR=PRODUSE\_PROIECT.COD\_FURNIZOR

AND PRODUSE\_PROIECT.CANTITATE\*PRODUSE\_PROIECT.PRET BETWEEN 100 AND 700

ORDER BY VALOARE DESC;

Table

Description automatically generated

13.Să se afişeze ID\_ANGAJAT,NUME,PRENUME angajatilor chiar daca acestia nu au contribuit la nicio comanda

SELECT a.ID\_ANGAJAT,a.NUME, a.PRENUME, c.COD\_COMANDA, c.NUME\_PRODUSE

FROM ANGAJATI\_PROIECT a, COMENZI\_PROIECT c

WHERE a.ID\_ANGAJAT = c.ID\_ANGAJAT(+);

Table

Description automatically generated

14.Sa se afiseze numele fiecarui angajat si numele cui i se subordoneaza(cod\_superior)

SELECT s.NUME||' i se subordoneaza lui : '||a.NUME

FROM ANGAJATI\_PROIECT a, ANGAJATI\_PROIECT s

WHERE a.COD\_FUNCTIE=s.COD\_SUPERIOR;

Text

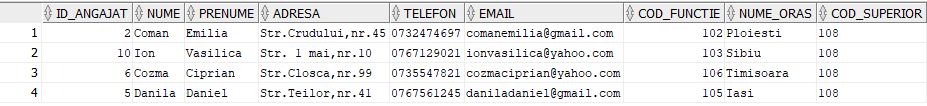
Description automatically generated

15.Sa se afiseze angajatii care au acelasi superior ca si Danila Daniel.

SELECT\*FROM ANGAJATI\_PROIECT

WHERE COD\_SUPERIOR IN

(SELECT COD\_SUPERIOR FROM ANGAJATI\_PROIECT WHERE NUME='Danila');



16.Să se afişeze numele furnizorilor care distribui produsele ce au preţul unitar cel mai mic.

SELECT f.NUME\_FURNIZOR, p.DENUMIRE, p.PRET

FROM FURNIZOR\_PROIECT f, PRODUSE\_PROIECT p

WHERE f.COD\_FURNIZOR=p.COD\_FURNIZOR

AND p.PRET=(SELECT MIN(PRODUSE\_PROIECT.PRET)FROM PRODUSE\_PROIECT);

A picture containing text

Description automatically generated

* **UTILIZAREA FUNCTIILOR PREDEFINITE IN INTEROGARI**

1.Sa se afiseze cu litere mari numele angajatilor cu cod\_superior=101

SELECT UPPER (NUME), UPPER(PRENUME),COD\_SUPERIOR

FROM ANGAJATI\_PROIECT

WHERE COD\_SUPERIOR=101;

Graphical user interface, text, application

Description automatically generated

2.Sa se afiseze angajatii al caror nume incepe cu litera C.

SELECT NUME,PRENUME FROM ANGAJATI\_PROIECT WHERE NUME LIKE UPPER('c%');

Graphical user interface, table

Description automatically generated with medium confidence

3.Sa se afiseze toti angajatii cu numele Stoica.

SELECT NUME,PRENUME FROM ANGAJATI\_PROIECT WHERE INITCAP(NUME)='Stoica';

SELECT NUME,PRENUME FROM ANGAJATI\_PROIECT WHERE UPPER(NUME)='STOICA';

SELECT NUME,PRENUME FROM ANGAJATI\_PROIECT WHERE LOWER(NUME)='stoica';

Table

Description automatically generated with medium confidence

4.Sa se afiseze denumirea produsului si cantitatea comandata.

SELECT 'Produsul: ' || DENUMIRE || ' este in cantitate de ' || CANTITATE cantitatea\_produsului

FROM PRODUSE\_PROIECT;

Text

Description automatically generated with medium confidence

Text

Description automatically generated

5.Sa se afiseze id\_angajati, numele angajatilor concatenata cu codul functiei acestora si lungimea prenumelui, nivel\_venituri numai pentru angajatii cu codul superiorului care incepe cu 1

SELECT ID\_ANGAJAT, CONCAT(NUME,COD\_FUNCTIE), LENGTH(PRENUME), COD\_SUPERIOR

FROM ANGAJATI\_PROIECT

WHERE SUBSTR(COD\_SUPERIOR,1,1)='1';

Table

Description automatically generated

6.Sa se afișeze numărul 154,99983 rotunjit la două zecimale si rotunjit la numar intreg.Sa se aplice si functia TRUNC.

SELECT ROUND(154.99983,2), ROUND(154.99983,0) FROM DUAL;



SELECT TRUNC(154.99983,2), TRUNC(154.99983,0) FROM DUAL;



7.Sa se afiseze perioada de timp corespunzatoare (în săptămâni) între data angajarii și data curentă.

SELECT ID\_ANGAJAT,(SYSDATE-DATA\_ANGAJARII)/7 saptamani

FROM ANGAJATI\_PROIECT;

Table

Description automatically generated with low confidence

8.Afisati data curenta (se selecteaza data din tabela DUAL):

SELECT SYSDATE data\_curenta FROM DUAL;



9.Sa se afiseze angajatii, data angajarii , numarul de luni intre data curenta si data angajarii, urmatoare zi de joi dupa data angajrii, ultima zi din luna in care face parte data incheierii precum si data corespunzatoare dupa 3 luni de la data angajarii.

SELECT ID\_ANGAJAT, DATA\_ANGAJARII,

ROUND(MONTHS\_BETWEEN(SYSDATE, DATA\_ANGAJARII)) luni,

NEXT\_DAY(DATA\_ANGAJARII, 'THURSDAY'),

LAST\_DAY(DATA\_ANGAJARII),

ADD\_MONTHS(DATA\_ANGAJARII,3)

FROM ANGAJATI\_PROIECT;

Graphical user interface

Description automatically generated with medium confidence

10.Sa se afiseze angajatii angajati luna trecuta.

SELECT NUME, PRENUME, DATA\_ANGAJARII

FROM ANGAJATI\_PROIECT

WHERE ROUND(MONTHS\_BETWEEN(sysdate, DATA\_ANGAJARII))=1;

Graphical user interface, application

Description automatically generated

11.Sa se afiseze angajatii angajati in 2015.Se va rotunji data angajarii la prima zi din luna corespunzătoare dacă data încheierii este în prima jumatate a lunii sau la prima zi din luna următoare:

SELECT NUME,PRENUME, DATA\_ANGAJARII, ROUND(DATA\_ANGAJARII, 'MONTH')

FROM ANGAJATI\_PROIECT

WHERE DATA\_ANGAJARII LIKE '%-15%';

Table

Description automatically generated

12.Sa se afiseze angajatii si data angajarii in formatul initial si in formatul "MM/YY".

SELECT NUME, PRENUME, DATA\_ANGAJARII, TO\_CHAR(DATA\_ANGAJARII,'MM/YY') data\_angajare

FROM ANGAJATI\_PROIECT;

Table

Description automatically generated

13.Sa se afiseze angajatii angajati intre 1 ianuarie 2020 si 31 decembrie 2021

SELECT NUME,PRENUME,DATA\_ANGAJARII

FROM ANGAJATI\_PROIECT

WHERE DATA\_ANGAJARII BETWEEN TO\_DATE('January 01, 2020','Month dd,YYYY') AND

TO\_DATE('December 31, 2021','Month dd, YYYY');

Graphical user interface, text, application

Description automatically generated

14.Sa se afiseze angajatii angajati in anii 2007 si 2009

SELECT NUME, PRENUME,DATA\_ANGAJARII

FROM ANGAJATI\_PROIECT

WHERE EXTRACT (YEAR FROM DATA\_ANGAJARII) IN (2007,2009);

Graphical user interface

Description automatically generated

15.Sa se afiseze angajatii angajati in lunile decembrie,ianurie

SELECT NUME,PRENUME,DATA\_ANGAJARII

FROM ANGAJATI\_PROIECT

WHERE EXTRACT(MONTH FROM DATA\_ANGAJARII) IN (1,12);

Graphical user interface, application

Description automatically generated

16.Sa se calculeze veniturile anuale ale angajatilor

SELECT NUME,PRENUME,SALARIU\*12+SALARIU\*12\*NVL(COMISION,0) VENIT\_ANUAL

FROM ANGAJATI\_PROIECT;

A picture containing text, crossword puzzle, receipt

Description automatically generated

17.Sa se afiseze angajatii care au comision (1) si pe cei care nu au comision (0).

SELECT NUME, PRENUME,NVL2(COMISION,1,0)

FROM ANGAJATI\_PROIECT;

Table

Description automatically generated

18.Sa se afiseze lungimea numelui, lungimea prenumelui, daca acestea sunt egale sa se returneze nul ca rezultat, iar daca nu sunt egale se va returna lungimea numelui.

SELECT NUME, LENGTH(NUME), PRENUME, LENGTH(PRENUME),

NULLIF(LENGTH(NUME),LENGTH(PRENUME))REZULTAT

FROM ANGAJATI\_PROIECT;

Table

Description automatically generated

19.Sa se afiseze cod superiorul fiecarui angajat, daca acesta este nul, se va afisa comisionul, iar daca si acesta este nul se va afisa -1.

SELECT NUME,PRENUME,COALESCE(COD\_SUPERIOR,COMISION,-1)

FROM ANGAJATI\_PROIECT;

Table

Description automatically generated

* **FUNCTIILE DE GRUP**

1.Să se afişeze valoarea maximă, valoarea medie, valoarea minimă şi valoarea totală a produselor comandate.

SELECT

MAX(p.CANTITATE\*P.PRET),AVG(p.CANTITATE\*P.PRET),MIN(p.CANTITATE\*P.PRET),SUM(p.CANTITATE\*P.PRET)

FROM PRODUSE\_PROIECT p;



2.Sa se afiseze data angajarii primului angajat si data angajarii ultimului angajat.

SELECT MIN(DATA\_ANGAJARII),MAX(DATA\_ANGAJARII)

FROM ANGAJATI\_PROIECT;



3.Sa se afiseze numarul de produse al caror pret<100.

SELECT COUNT(\*) nr\_produse

FROM PRODUSE\_PROIECT

WHERE PRET<100;



4.Sa se afiseze numarul de produse care au pretul distinct.

SELECT COUNT(PRET) FROM PRODUSE\_PROIECT;



SELECT COUNT(DISTINCT PRET) FROM PRODUSE\_PROIECT;



5.Sa se afiseze numarul total de comenzi incheiate.

SELECT COUNT(COD\_COMANDA) NR\_COMENZI FROM COMENZI\_PROIECT;



6.Sa se afiseze cate produse se afla pe comanda cu numarul 888.

SELECT COUNT (COD\_COMANDA) produse\_comandate

FROM PRODUSE\_PROIECT WHERE COD\_COMANDA=888;



7.Să se afişeze cantitatea medie vândută din fiecare produs. Sa se ordoneze după cantitate (se utilizeaza functia AVG() si clauza GROUP BY pentru gruparea datelor in functie de numele produsului, iar ordonarea se realizeaza cu ajutorul functiei ORDER BY).

SELECT denumire, ROUND(AVG(cantitate),2) medie\_produse

FROM produse\_proiect

GROUP BY denumire

ORDER BY medie\_produse;

Graphical user interface, application, table

Description automatically generated

8.Să se afişeze produsele şi cantitatea medie vândută numei pentru acele produse a căror cantitate medie este mai mare de 25 (conditia se specifica in clauza HAVING si nu in clauza WHERE deoarece este utilizata functia de grup AVG si conditia este AVG(cantitate)>10).

SELECT denumire, ROUND(AVG(cantitate),1) medie\_produse

FROM produse\_proiect

GROUP BY denumire

HAVING ROUND(AVG(cantitate),1)>10;

Table

Description automatically generated

9.Sa se calculeze valoarea totala a fiecarei comenzi si sa se sorteze descrescator in functie de valoare

SELECT c.cod\_comanda, SUM(p.cantitate \* p.pret) total\_comanda

FROM comenzi\_proiect c, produse\_proiect p

WHERE p.cod\_comanda=c.cod\_comanda

GROUP BY c.cod\_comanda

ORDER BY total\_comanda DESC;

Table

Description automatically generated

10.Sa se afiseze numai comenzile care au valoarea cuprinsa intre 1000 si 5000(conditia va fi mentionata in clauza HAVING deoarece se utilizeaza functia de grup SUM):

SELECT c.cod\_comanda, SUM(p.cantitate \* p.pret) total\_comanda

FROM comenzi\_proiect c, produse\_proiect p

WHERE p.cod\_comanda=c.cod\_comanda

GROUP BY c.cod\_comanda

HAVING SUM (p.cantitate \* p.pret) BETWEEN 1000 AND 5000

ORDER BY total\_comanda DESC;

Table

Description automatically generated

* **PARCURGEREA STRUCTURILOR IERARHICE**

1. Parcurgerea arborelui TOP-BOTTOM

1.Sa se afiseze angajatii si nivelul ierarhic al acestora pornind de la angajatul cod\_functie =101 (sa se ordoneze in functie de nivelul ierahic).

SELECT cod\_functie, nume,prenume, cod\_superior, LEVEL FROM angajati\_proiect

CONNECT BY PRIOR cod\_functie=cod\_superior

START WITH cod\_functie=101

ORDER BY LEVEL;

Table

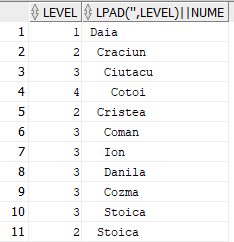
Description automatically generated

2.Afisati angajatii companiei subordonati inregistrarii radacina sub forma de organigrama.

SELECT LEVEL, LPAD(' ', LEVEL)|| nume FROM angajati\_proiect

CONNECT BY PRIOR cod\_functie=cod\_superior

START WITH cod\_functie=101;



3.Afisati angajatii companiei subordonati inregistrarii radacina specificand numarul de superiori si toti superiorii sai, cod\_functie(se utilizeaza clauzele: SYS\_CONNECT\_BY\_PATH, LEVEL-1).

SELECT cod\_functie, nume,

LEVEL-1 Numar\_Superiori, SYS\_CONNECT\_BY\_PATH(cod\_functie, '/') ID\_Superiori

FROM angajati\_proiect

START WITH cod\_functie=101

CONNECT BY PRIOR cod\_functie=cod\_superior;

Table

Description automatically generated

Interogari ierarhice conditionate (clauza WHERE):

4.Sa se selecteze angajati si gradul de subordonare numai pentru cei din departamentele 20 si 50:

SELECT cod\_functie, nume, cod\_superior,salariu, level FROM angajati\_proiect

WHERE salariu BETWEEN 2000 AND 6000

CONNECT BY PRIOR cod\_functie=cod\_superior

START WITH cod\_functie=101;

Table

Description automatically generated

1. Parcurgerea arborelui BOTTOM UP

5.Sa se afiseze superiorii angajatilor aflati pe ultimul nivel de subordonare.

SELECT nume, LEVEL-1 Numar\_Superiori, SYS\_CONNECT\_BY\_PATH(nume, '/') Nume\_Superiori

FROM angajati\_proiect

START WITH cod\_superior = 101

CONNECT BY PRIOR cod\_functie=cod\_superior

ORDER BY LEVEL desc;

A screenshot of a computer

Description automatically generated with medium confidence

6.Sa se afiseze toti superiorii lui ‘Coman’.

SELECT cod\_functie, nume, cod\_superior, LEVEL FROM angajati\_proiect

CONNECT BY cod\_functie = PRIOR cod\_superior

START WITH nume= 'Coman';

Table

Description automatically generated

1. Parcurgerea arborelui incepind cu un anumit nivel

7.Sa se afiseze toti subordonatii lui 'Daia'.

SELECT cod\_functie, nume, cod\_superior, LEVEL FROM angajati\_proiect

CONNECT BY PRIOR cod\_functie=cod\_superior

START WITH nume= 'Daia'

ORDER BY LEVEL;

Table

Description automatically generated

8. Sa se afiseze toti superiorii lui 'Ciutacu'.

SELECT cod\_functie, nume, cod\_superior, LEVEL FROM angajati\_proiect

CONNECT BY cod\_functie=PRIOR cod\_superior

START WITH nume= 'Ciutacu'

ORDER BY LEVEL;

Table

Description automatically generated

9. Sa se afiseze toti subordonatii lui 'Ciutacu' si care au acelasi salariu ca 'Craciun'.

SELECT cod\_functie, nume, cod\_superior,salariu, LEVEL FROM angajati\_proiect

WHERE salariu = (SELECT salariu FROM angajati\_proiect WHERE nume='Ciutacu')

CONNECT BY PRIOR cod\_functie=cod\_superior

START WITH nume = 'Craciun'

ORDER BY LEVEL;

Table

Description automatically generated

10.Sa se selecteze toti angajatii cu salariu=2340 si subordonatii acestora.

SELECT cod\_functie, nume, cod\_superior,salariu, LEVEL FROM angajati\_proiect

CONNECT BY PRIOR cod\_functie=cod\_superior

START WITH salariu=2340

ORDER BY LEVEL;

Table

Description automatically generated

11.Sa se afiseze toti subordonatii lui 'Daia' fara cei cu salariu <1000.

SELECT cod\_functie, nume, cod\_superior,salariu, LEVEL FROM angajati\_proiect

WHERE salariu>2000

CONNECT BY PRIOR cod\_functie=cod\_superior

START WITH nume= 'Daia'

ORDER BY LEVEL;

Table

Description automatically generated

12.Sa se afiseze toti subordonatii lui 'Daia' si mai putin cei cu salariul 2340,3240.

SELECT cod\_functie, nume, cod\_superior,salariu, LEVEL FROM angajati\_proiect

WHERE salariu NOT IN (2340,3240)

CONNECT BY PRIOR cod\_functie=cod\_superior

START WITH nume= 'Daia'

ORDER BY LEVEL;

Table

Description automatically generated

13.Sa se afiseze toti subordonatii lui 'Cotoi' si cei ai lui ‘Cristea'.

SELECT cod\_functie, nume, cod\_superior, LEVEL FROM angajati\_proiect

CONNECT BY PRIOR cod\_functie=cod\_superior

START WITH nume = 'Cotoi' OR nume= 'Cristea'

ORDER BY LEVEL;

Table

Description automatically generated

14.Sa se afiseze toti subordonatii de pe nivelul 3.

SELECT cod\_functie, nume, cod\_superior, LEVEL FROM angajati\_proiect

WHERE LEVEL=3

CONNECT BY PRIOR cod\_functie=cod\_superior

ORDER BY LEVEL;

Table

Description automatically generated

15.Sa se afiseze toti subordonatii incepand cu nivelul 3.

SELECT cod\_functie, nume, cod\_superior, LEVEL FROM angajati\_proiect

WHERE LEVEL>=3

CONNECT BY PRIOR cod\_functie=cod\_superior

ORDER BY LEVEL;

Table

Description automatically generated

16.Sa se afiseze numarul total de angajati subordonati lui 'Steven King' grupati pe salarii.

SELECT salariu, count(\*) Total\_Angajati from angajati\_proiect

CONNECT BY PRIOR cod\_functie=cod\_superior

START WITH cod\_functie=101

GROUP BY salariu;

Table

Description automatically generated

17.Sa se selecteze toti subordonatii angajatilor cu emailul gmail

SELECT cod\_functie, nume, cod\_superior,email FROM angajati\_proiect

CONNECT BY PRIOR cod\_functie=cod\_superior

START WITH email LIKE '%gmail%'

ORDER BY LEVEL;

Table

Description automatically generated

* **Jonctiuni externe**

**Functia DECODE si expresia CASE**

**1) Jonctiuni externe:**

1. Sa se afiseze comenzile efecuate, chiar daca nu exista produse plasate in comanda respective.

SELECT c.cod\_comanda,c.pret\_total, p.cod\_articol,p.denumire

FROM produse\_proiect p, comenzi\_proiect c

WHERE c.cod\_comanda = p.cod\_comanda(+);

Table

Description automatically generatedTable

Description automatically generated

2. Sa se afiseze ID\_ANGAJAT,NUME,PRENUME angajatilor chiar daca acestia nu au contribuit la nicio comanda.

SELECT a.ID\_ANGAJAT,a.NUME, a.PRENUME, c.COD\_COMANDA, c.NUME\_PRODUSE

FROM ANGAJATI\_PROIECT a, COMENZI\_PROIECT c

WHERE a.ID\_ANGAJAT = c.ID\_ANGAJAT(+);

Graphical user interface, text, application

Description automatically generated

2) Realizarea interogarilor pe baza unor conditii

1.Sa se identifice ierarhia angajatilor cu emailul 'gmail'.

SELECT id\_angajat, nume,email, LEVEL FROM angajati\_proiect

CONNECT BY cod\_functie=PRIOR cod\_superior

START WITH email like '%gmail%';

Table

Description automatically generated

2.Sa se calculeze comisionul angajatilor care se ocupa de comenzi in functie de pozitia (functia) ocupata:

•0.2% din valoarea comenzilor daca functia este sef depozit

•0.3% din valoarea comenzilor daca functia este sef tura

•0.5% din valoarea comenzilor daca functia este gestionar magazin

Pentru celelalte functii comisionul va fi 0.

SELECT a.nume, a.cod\_functie,

(CASE WHEN UPPER(a.cod\_functie) = 108 THEN 0.3

WHEN UPPER(a.cod\_functie)= 105 THEN 0.2

WHEN UPPER(a.cod\_functie)= 109 THEN 0.5

ELSE 0 END)\*SUM(p.cantitate\*p.pret) valoare\_comision

FROM angajati\_proiect a, comenzi\_proiect c, produse\_proiect p

WHERE a.id\_angajat=c.id\_angajat

AND c.cod\_comanda=p.cod\_comanda

GROUP BY a.nume, a.cod\_functie;

Table

Description automatically generated with low confidence

-Cu functia DECODE:

SELECT nume, cod\_functie,

DECODE(cod\_functie , 108 , 0.3,105 , 0.2, 109 , 0.5, 0) comision

FROM angajati\_proiect;

Table

Description automatically generated

3.Sa se calculeze bonusul angajatilor in functie de cate comnezi au efectuat.

•daca angajatul a incheiat 1 comanda atunci bonus= 10%;

•daca a incheiat 2 comenzi atunci bonus =15%;

•daca a incheiat mai mult de 3 comenzi atunci bonus =20%.

SELECT a.nume, count(c.id\_angajat) numar\_comenzi,

CASE WHEN COUNT(c.id\_angajat) = 1 THEN 0.1

WHEN COUNT(c.id\_angajat) = 2 THEN 0.15

WHEN COUNT(c.id\_angajat)>=3 THEN 0.2

ELSE 0 END bonus

FROM angajati\_proiect a, comenzi\_proiect c

WHERE a.id\_angajat=c.id\_angajat

GROUP BY a.nume;

Table

Description automatically generated

* Operatorii algebrei relationale: UNION, INTERSECT, MINUS

3.1) Operatorul MINUS

1. Sa se afiseze angajatii care au salariul intre 2000 si 6000 fara cei care au salariul 3240 si 5400.

SELECT \* FROM angajati\_proiect WHERE salariu BETWEEN 2000 AND 6000

MINUS

SELECT \* FROM angajati\_proiect WHERE salariu IN (3240, 5400);

Graphical user interface

Description automatically generated with medium confidence

2. Sa se calculeze diferit bonusul pentru angajati astfel:

•daca angajatul a incheiat 1 comanda atunci bonus= 10%;

•daca a incheiat 2 comenzi atunci bonus =15%;

•daca a incheiat mai mult de 3 comenzi atunci bonus =20%.

Din acestea sa se elimine inregistrarile incheiate de angajatii care incep cu litera C.

Ordonati crescator in functie de numele clientilor.

SELECT a.nume, COUNT(c.id\_angajat) numar\_comenzi,

(CASE WHEN COUNT(c.id\_angajat)=1 THEN 0.1

WHEN COUNT(c.id\_angajat)=2 THEN 0.15

WHEN COUNT(c.id\_angajat)>=3 THEN 0.2

ELSE 0 END) bonus

FROM angajati\_proiect a, comenzi\_proiect c

WHERE a.id\_angajat=c.id\_angajat

GROUP BY a.nume

MINUS

SELECT a.nume, COUNT(c.id\_angajat) numar\_comenzi,

(CASE WHEN COUNT(c.id\_angajat)=1 THEN 0.1

WHEN COUNT(c.id\_angajat)=2 THEN 0.15

WHEN COUNT(c.id\_angajat)>=3 THEN 0.2

ELSE 0 END) bonus

FROM angajati\_proiect a, comenzi\_proiect c

WHERE a.id\_angajat=c.id\_angajat

AND a.nume like 'C%'

GROUP BY a.nume

ORDER BY nume;

A picture containing text

Description automatically generated

3.2.) Operatorul UNION

3.Sa se calculeze comisionul angajatilor care se ocupa de comenzi in functie de pozitia (functia) ocupata, folosind operatorul UNION.

•0.2% din valoarea comenzilor daca functia este sef depozit

•0.3% din valoarea comenzilor daca functia este sef tura

•0.5% din valoarea comenzilor daca functia este gestionar magazine

SELECT a.nume, COUNT(c.id\_angajat) numar\_comenzi,

0.2\* SUM(p.cantitate\*p.pret) valoare\_comision

FROM angajati\_proiect a, comenzi\_proiect c, produse\_proiect p

WHERE a.id\_angajat=c.id\_angajat

AND c.cod\_comanda=p.cod\_comanda

GROUP BY a.nume

HAVING COUNT(c.id\_angajat)=1

UNION

SELECT a.nume, COUNT(c.id\_angajat) numar\_comenzi,

0.3\* SUM(p.cantitate\*p.pret) valoare\_comision

FROM angajati\_proiect a, comenzi\_proiect c, produse\_proiect p

WHERE a.id\_angajat=c. id\_angajat

AND c.cod\_comanda=p.cod\_comanda

GROUP BY a.nume

HAVING COUNT(c.id\_angajat)=2

UNION

SELECT a.nume, COUNT(c.id\_angajat) numar\_comenzi,

0.5\* SUM(p.cantitate\*p.pret) valoare\_comision

FROM angajati\_proiect a, comenzi\_proiect c, produse\_proiect p

WHERE a.id\_angajat=c. id\_angajat

AND c.cod\_comanda=p.cod\_comanda

GROUP BY a.nume

HAVING COUNT(c.id\_angajat)>=3;

Graphical user interface

Description automatically generated with medium confidence

3.3.) Operatorul INTERSECT

Sa se selecteze numele,numarul de comenzi pentru toti angajatii care au efectuat cel putin 1 comanda si care au salariul minim 2340.

SELECT a.nume,COUNT(c.id\_angajat) numar\_comenzi

FROM angajati\_proiect a, comenzi\_proiect c

WHERE a.id\_angajat=c.id\_angajat

GROUP BY a.nume

HAVING COUNT(c.id\_angajat)>=1

INTERSECT

SELECT a.nume,COUNT(c.id\_angajat) numar\_comenzi

FROM angajati\_proiect a, comenzi\_proiect c

WHERE a.id\_angajat=c.id\_angajat

GROUP BY a.nume

HAVING MIN(a.salariu)=2340;

Table

Description automatically generated

GESTIUNEA ALTOR OBIECTE ALE BAZEI DE DATE

* TABELE VIRTUALE

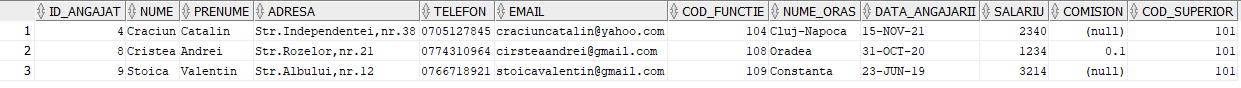
1.Sa realizeze o tabela virtuala cu toti angajatii cu cod superior=101. Actualizam salariul(-1000).

CREATE OR REPLACE VIEW v\_angajati\_101

AS SELECT \* FROM angajati\_proiect

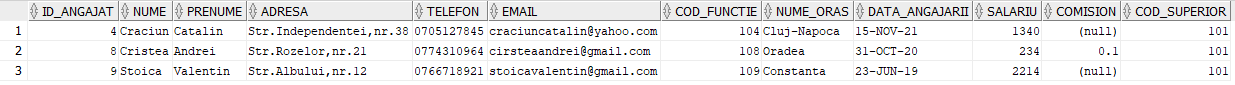
WHERE cod\_superior=101;

SELECT \* FROM v\_angajati\_101;



UPDATE v\_angajati\_101

SET salariu = salariu-1000;



2.Stocarea unei interogari care sa permita adaugarea unor conditii ulterioare.

SELECT \* FROM v\_angajati\_101 WHERE EXTRACT(MONTH FROM data\_angajarii) = 10;



3.Actualizarea tabelelor virtuale.

CREATE OR REPLACE VIEW v\_produse

AS SELECT cod\_articol, pret, pret\*cantitate valoare FROM produse\_proiect;

A picture containing chart

Description automatically generated

UPDATE v\_produse

SET valoare = valoare+5;

A picture containing chart

Description automatically generated

4.Opţiunea WITH READ ONLY

CREATE OR REPLACE VIEW v\_produse

AS SELECT cod\_articol, pret, pret\*cantitate valoare FROM produse\_proiect

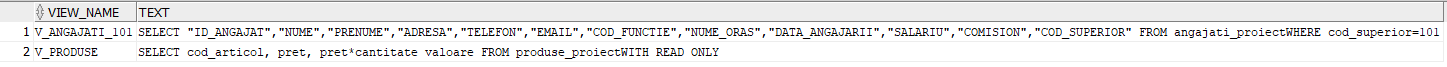
WITH READ ONLY;

5. Sa se stearga tabela virtuală v\_proiect.

DROP VIEW v\_produse;

6.Vizualizarea informatiilor despre tabelele virtuale.

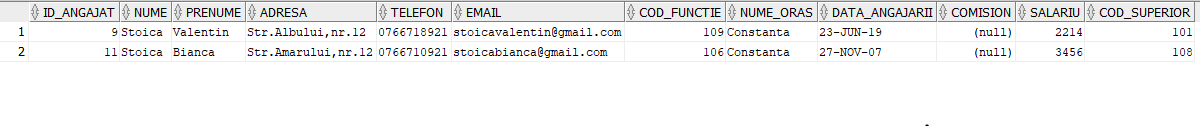
SELECT VIEW\_NAME, TEXT FROM USER\_VIEWS;



* INDECȘI

1.Sa se creeze un index pe tabela angajati pe coloana nume.

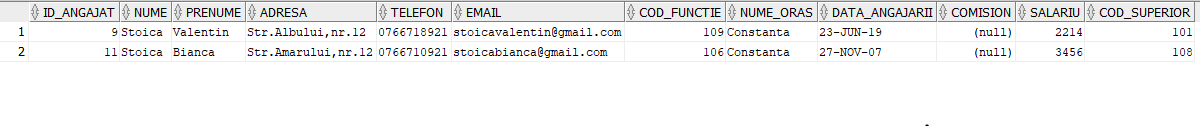
SELECT \* FROM angajati\_proiect WHERE nume = 'Stoica';



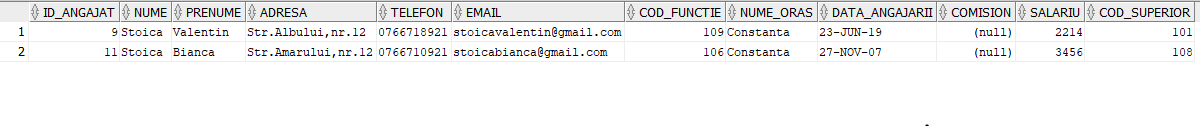
CREATE INDEX idx\_nume\_proiect ON angajati\_proiect(nume);



SELECT \* FROM angajati\_proiect WHERE nume = 'Stoica';



SELECT \* FROM angajati\_proiect WHERE UPPER(nume) = 'STOICA';

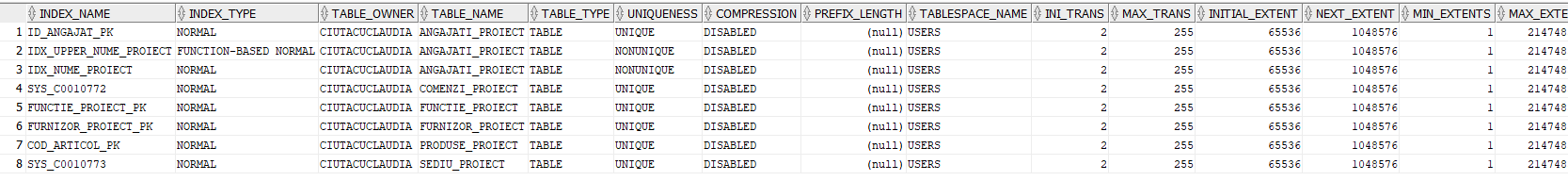


CREATE INDEX idx\_upper\_nume\_proiect ON angajati\_proiect(UPPER(nume));



2.Vizualizarea indecşilor unui anumit utilizator.

SELECT \* FROM USER\_INDEXES;



3. Sa se stearga indexul creat anterior.

DROP INDEX idx\_nume\_proiect;

* SECVENȚE

1. Sa se creeze o secventa pentru asigurarea unicitatii cheii primare din tabela comenzi\_proiect.

CREATE SEQUENCE seq\_codcomanda

START WITH 1 INCREMENT BY 1

MAXVALUE 1000 NOCYCLE;

Graphical user interface, application, table

Description automatically generated

INSERT INTO comenzi\_proiect VALUES (seq\_codcomanda.NEXTVAL,'cutit',20,5);

Graphical user interface, text, application, email

Description automatically generated

2.Sa se afiseze valoarea curenta a secventei.

SELECT seq\_codcomanda.CURRVAL FROM DUAL;

Text

Description automatically generated with medium confidence

3.Să se modifice pasul de incrementare şi valoarea maximă pentru secvenţa anterioară.

ALTER SEQUENCE seq\_codcomanda INCREMENT BY 100;

ALTER SEQUENCE seq\_codcomanda MAXVALUE 3000;

INSERT INTO comenzi\_proiect VALUES (seq\_codcomanda.NEXTVAL,'cutit',20,5);

Graphical user interface, text, application, email

Description automatically generated

4.Să se vizualizeze informaţiile depre secvenţele utilizatorilor.

SELECT \* FROM USER\_SEQUENCES;



5.Să se steargă secvenţa seq\_codcomanda.

DROP SEQUENCE seq\_codcomanda;

* SINONIME

1.Sa se creeze un sinonim pentru tabela sediu\_proiect

CREATE SYNONYM sp FOR FURNIZOR\_PROIECT;

2.Vizualizarea sinonimelor se realizeaza astfel.

SELECT \* FROM USER\_SYNONYMS;



3.Sa se stearga sinonimul creat anterior.

DROP SYNONYM sp;

* PARTIȚII

CREATE TABLE tabela\_proiect(data\_angajarii DATE, nume VARCHAR2(20), prenume VARCHAR2(20))

PARTITION BY RANGE(data\_angajarii)

(PARTITION P1 VALUES LESS THAN (TO\_DATE('17.11.2021','DD.MM.YYYY')),

PARTITION P2 VALUES LESS THAN (TO\_DATE('17.12.2021','DD.MM.YYYY')));

INSERT INTO tabela\_proiect SELECT data\_angajarii, nume, prenume FROM angajati\_proiect;

SELECT \* FROM tabela\_proiect WHERE data\_angajarii <TO\_DATE('01.02.2019','DD.MM.YYYY');

Table

Description automatically generated

SELECT \* FROM tabela\_proiect partition (p1) WHERE data\_angajarii <TO\_DATE('01.02.2021','DD.MM.YYYY');

Table

Description automatically generated

SELECT \* FROM angajati\_proiect WHERE data\_angajarii <TO\_DATE('01.02.2021','DD.MM.YYYY');

Graphical user interface, text, application

Description automatically generated

* CLUSTERE

Sa se realizeze un cluster pentru campul cod\_comanda si sa se adauge la acest cluster 2 tabele: comenzi2 si produse2 cu aceeasi structura cu tabelele din aplicatie.

CREATE CLUSTER cls\_codcomanda (cod\_comanda NUMBER(5,0));

CREATE INDEX idx\_codcomanda ON CLUSTER cls\_codcomanda;

CREATE TABLE comenzi2

CLUSTER cls\_codcomanda (cod\_comanda)

AS SELECT \* FROM comenzi\_proiect;

Graphical user interface

Description automatically generated

CREATE TABLE produse2

CLUSTER cls\_codcomanda (cod\_comanda)

AS SELECT \* FROM produse\_proiect;

Table

Description automatically generated

Vizualizarea informatiilor despre clustere:

SELECT \* FROM USER\_CLUSTERS;

