



ACADEMIA DE STUDII ECONOMICE  
FACULTATEA DE CIBERNETICA, STATISTICA SI INFORMATICA ECONOMICA

## Proiect disciplina “Baze de date”

Gestiunea datelor dintr-o cofetarie

**Profesor coordonator:**

CAUNIAC Diana Andreea

**Proiect realizat de:**

MORARU Marielena-Lorena

Grupa 1059, Seria D, An II

# **Cuprins**

1. Obiectivul proiectului
  - 1.2. Schema bazei de date
  - 1.3. Crearea tabelelor
2. Comenzi folosind limbajul de manipulare a datelor: INSERT, UPDATE, DELETE, MERGE.
  - 2.1. Adaugarea inregistrarilor
  - 2.2. Actualizarea inregistrarilor
  - 2.3. Stergerea si recuperarea unei tabele
3. Exemple de interogari variate
4. Gestiunea altor obiecte ale bazei de date

- **CERINTE PROIECT ORACLE**

1. Definirea schemei bazei de date – tabele, restrictii de integritate. Se utilizeaza comenzi CREATE, ALTER, DROP.
2. Exemple cu operatiile de actualizare a datelor (comenzi DML – INSERT, UPDATE, DELETE, MERGE (optional) pentru inregistrari).
3. Exemple de interogari variate (SELECT):
  - Utilizarea operatorilor de comparatie
  - Join-uri
  - Utilizarea functiilor de grup si conditii asupra acestora
  - Utilizarea functiilor numerice, de tip caracter, pentru data si timp
  - Construirea de expresii cu DECODE si CASE
  - Utilizarea operatorilor UNION, MINUS, INTERSECT
  - Subcereri (cereri imbricate)
4. Gestiunea altor obiecte ale bazei de date: vederi, indecsi, sinonime, sevante etc.

## **1. Obiectivul proiectului**

Baza de date ce urmeaza a fi prezentata are ca scop gestionarea eficientă a informațiilor legate de produse, ingrediente, rețete, comenzi, angajați, furnizori și materiale în cadrul unei cofetării. Scopul este să ofere o platformă centralizată pentru administrarea inventarului, proceselor de producție, comenziilor clienților, gestionării personalului și a materialelor.

### **Descrierea atributelor:**

Denumire Tabela	Atribute	Tip Data	Restrictii
Produse	ID_Produs Nume_Produs Tip Pret	NUMBER(3) VARCHAR(15) VARCHAR(10) NUMBER(5)	PK
Ingrediente	ID_Ingredient Nume_Ingredient	NUMBER(3) VARCHAR(15)	PK U

Reteta	ID_Reteta ID_Produs ID_Ingredient Cantitate	NUMBER(3) NUMBER(3) NUMBER(3) NUMBER(4)	PK FK FK
Comenzi	ID_Commanda ID_Produs Data_Commanda Cantitate Nume_Client AdLivrare	NUMBER(3) NUMBER(3) DATE NUMBER(3) VARCHAR(15) VARCHAR(15)	PK FK
Angajati	ID_Angajat Nume Prenume Functie Salariu	NUMBER(3) VARCHAR(15) VARCHAR(15) VARCHAR(10) NUMBER(5)	PK
Furnizori	ID_Furnizor Nume_Furnizor AdFurnizor	NUMBER(3) VARCHAR(20) VARCHAR(30)	PK
Materiale	ID_Material ID_Furnizor Nume_Material Pret_Unitar	NUMBER(3) NUMBER(3) VARCHAR(15) NUMBER(5, 2)	PK FK

### Relațiile dintre tabele:

- Tabelul Produse și Retete: Un produs poate avea mai multe rețete asociate, dar o rețetă este legată la un singur produs. (Tip: One-to-Many)
- Tabelul Ingrediente și Retete: Un ingredient poate apărea în mai multe rețete, dar o rețetă este legată la un singur ingredient. (Tip: One-to-Many)
- Tabelul Produse și Comenzi: Un produs poate apărea în mai multe comenzi, dar o comandă este legată la un singur produs. (Tip: One-to-Many)
- Tabelul Furnizori și Materiale: Un furnizor poate furniza mai multe materiale, dar un material este furnizat de un singur furnizor. (Tip: One-to-Many)
- 

### Descrierea tabelelor:

Tabela Produse - Această tabelă conține informații despre produsele disponibile în cofetărie. Fiecare produs are un identificator unic, nume, tip/categorie și preț.

Tabela Ingrediente - Această tabelă conține lista de ingrediente folosite în producția produselor cofetăriei. Fiecare ingredient are un identificator unic și un nume unic.

Tabela Retete - Această tabelă reprezintă asocierea dintre produse și ingrediente, specificând cantitatea necesară a fiecărui ingredient pentru a produce un anumit produs. Fiecare rețetă are un identificator unic și este legată la un produs și la un ingredient.

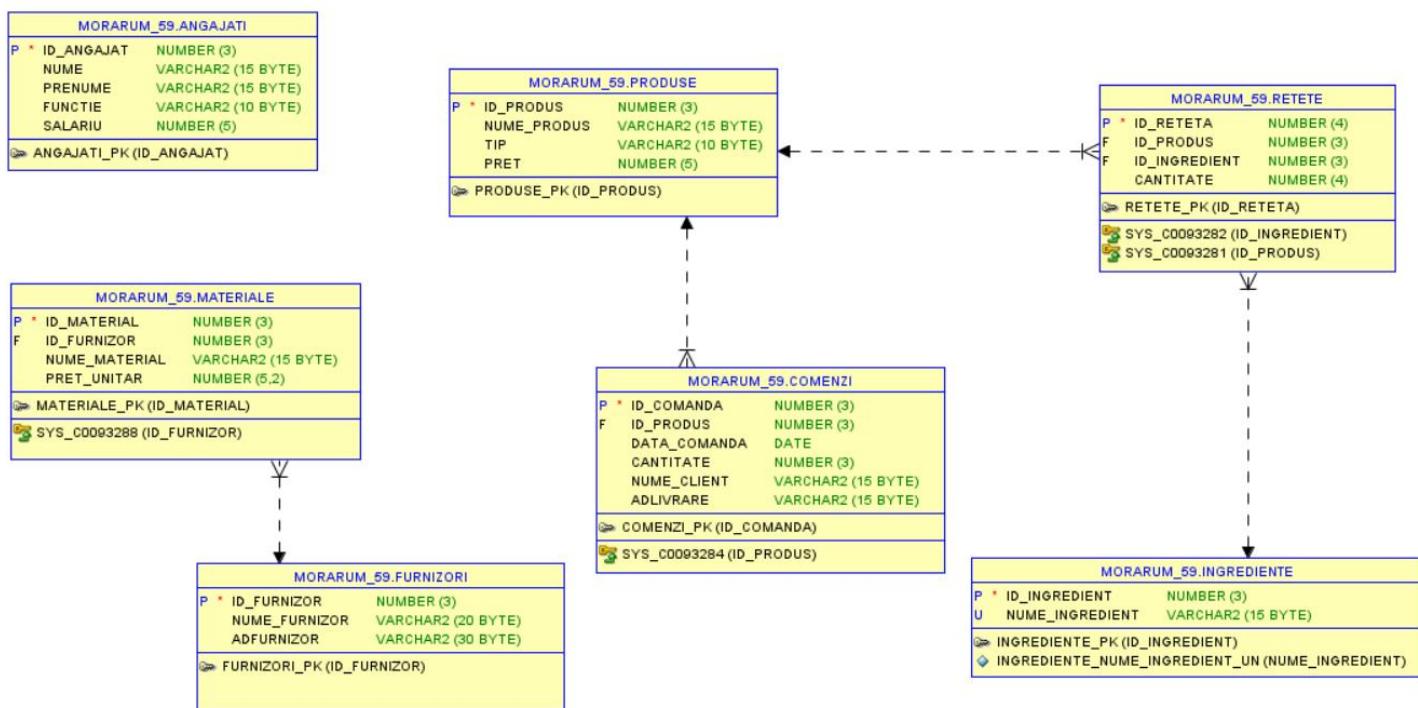
Tabela Comenzi - Această tabelă conține informații despre comenzi plasate de către clienți. Fiecare comandă are un identificator unic, produsul comandat, data comenzi, cantitatea, numele clientului și adresa de livrare.

Tabela Angajati - Această tabelă conține detalii despre angajații cofetăriei. Fiecare angajat are un identificator unic, nume, prenume, funcție și salariu.

Tabela Furnizori - Această tabelă conține informații despre furnizorii de materiale. Fiecare furnizor are un identificator unic, nume și adresă.

Tabela Materiale - Această tabelă conține detalii despre materialele furnizate de către furnizori. Fiecare material are un identificator unic, nume, preț unitar și este legat la un furnizor.

## 1.2. Schema bazei de date:



### 1.3. Crearea tabelelor

#### Tabela “Produse”

```
CREATE TABLE Produse (
    ID_Produs NUMBER(3),
    Nume_Produs VARCHAR(15),
    Tip VARCHAR(10),
    Pret NUMBER(5),
    Coloana_Drop VARCHAR(6)
);
```

```
ALTER TABLE Produse
ADD CONSTRAINT ID_PRODUS_PK primary key (ID_Produs);
ALTER TABLE Produse
DROP COLUMN Coloana_Drop;
```

The screenshot shows the Oracle SQL Developer interface. On the left, the Connections pane displays an Oracle connection named 'MoraruMarielena' with several tables listed under 'Tables': ANGAJATI, COMENZI, FURNIZORI, INGREDIENTE, MATERIALE, PRODUSE, and RETETE. The 'PRODUSE' table is selected and highlighted in blue. On the right, the main workspace shows the 'Columns' tab of the PRODUSE table editor. The table structure is as follows:

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_PRODUS	NUMBER(3,0)	No	(null)	1	(null)
2 NUME_PRODUS	VARCHAR2(15 BYTE)	Yes	(null)	2	(null)
3 TIP	VARCHAR2(10 BYTE)	Yes	(null)	3	(null)
4 PRET	NUMBER(5,0)	Yes	(null)	4	(null)

#### Tabela “Ingrediente”

```
CREATE TABLE Ingrediente (
    ID_Ingredient NUMBER(3),
    Nume_Ingredient VARCHAR(15)
);
```

```

ALTER TABLE Ingrediente
ADD CONSTRAINT ID_INGREDIENT_PK primary key (ID_Ingredient);

ALTER TABLE Ingrediente
ADD CONSTRAINT NUME_INGREDIENT_UQ unique (Nume_Ingredient);

```

The screenshot shows the Oracle SQL Developer interface. On the left, the 'Connections' sidebar shows an Oracle connection named 'MoraruMarielena'. Under 'Tables', the 'INGREDIENTE' table is selected. The main window displays the 'Columns' tab for the 'INGREDIENTE' table, listing two columns:

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
ID_INGREDIENT	NUMBER(3,0)	No	(null)	1	(null)
NUME_INGREDIENT	VARCHAR2(15 BYTE)	Yes	(null)	2	(null)

### Tabela “Retete”

```

CREATE TABLE Retete (
    ID_Reteta NUMBER(4),
    ID_Produs NUMBER(3),
    ID_Ingredient NUMBER(3),
    Cantitate NUMBER(4)
);

```

```

ALTER TABLE Retete
ADD CONSTRAINT ID_RETETA_PK primary key (ID_Reteta);

ALTER TABLE Retete
ADD CONSTRAINT ID_PRODUS_FK foreign key (ID_Produs) references Produse(ID_Produs);

ALTER TABLE Retete
ADD CONSTRAINT ID_INGREDIENT_FK foreign key (ID_Ingredient) references
Ingrediente(ID_Ingredient);

```

Connections

MoraruMarielena

Tables

- ANGAJATI
- COMENZI
- FURNIZORI
- INGREDIENTE
- MATERIALE
- PRODUSE
- RETETE

MoraruMarielena RETETE

Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Actions...

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_RETETA	NUMBER(4,0)	No	(null)	1	(null)
2 ID_PRODUS	NUMBER(3,0)	Yes	(null)	2	(null)
3 ID_INGREDIENT	NUMBER(3,0)	Yes	(null)	3	(null)
4 CANTITATE	NUMBER(4,0)	Yes	(null)	4	(null)

### Tabela “Comenzi”

```
CREATE TABLE Comenzi (
    ID_Comanda NUMBER(3),
    ID_Produs NUMBER(3),
    Data_Comanda DATE,
    Cantitate NUMBER(3),
    Nume_Client VARCHAR(15),
    AdLivrare VARCHAR(15)
);
```

```
ALTER TABLE Comenzi
```

```
ADD CONSTRAINT ID_COMANDA_PK primary key (ID_Comanda);
```

```
ALTER TABLE Comenzi
```

```
ADD CONSTRAINT ID_PRODUS2_FK foreign key (ID_Produs) references
Produse(ID_Produs);
```

Connections

MoraruMarielena

Tables

- ANGAJATI
- COMENZI
- FURNIZORI
- INGREDIENTE
- MATERIALE
- PRODUSE
- RETETE

MoraruMarielena COMENZI

Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes Actions...

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_COMANDA	NUMBER(3,0)	No	(null)	1	(null)
2 ID_PRODUS	NUMBER(3,0)	Yes	(null)	2	(null)
3 DATA_COMANDA	DATE	Yes	(null)	3	(null)
4 CANTITATE	NUMBER(3,0)	Yes	(null)	4	(null)
5 NUME_CLIENT	VARCHAR2(15 BYTE)	Yes	(null)	5	(null)
6 ADLIVRARE	VARCHAR2(15 BYTE)	Yes	(null)	6	(null)

### Tabela “Angajati”

```
CREATE TABLE Angajati (
```

```

ID_Angajat NUMBER(3),
Nume VARCHAR(15),
Prenume VARCHAR(15),
Functie VARCHAR(10),
Salariu NUMBER(5)
);

```

```

ALTER TABLE Angajati
ADD CONSTRAINT ID_ANGAJAT_PK primary key (ID_Angajat);

```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_ANGAJAT	NUMBER(3,0)	No	(null)	1	(null)
2 NUME	VARCHAR2(15 BYTE)	Yes	(null)	2	(null)
3 PRENUME	VARCHAR2(15 BYTE)	Yes	(null)	3	(null)
4 FUNCTIE	VARCHAR2(10 BYTE)	Yes	(null)	4	(null)
5 SALARIU	NUMBER(5,0)	Yes	(null)	5	(null)

#### Tabela "Furnizori"

```

CREATE TABLE Furnizori (
ID_Furnizor NUMBER(3),
Nume_Furnizor VARCHAR(20),
AdFurnizor VARCHAR(30)
);

```

```

ALTER TABLE Furnizori
ADD CONSTRAINT ID_FURNIZOR_PK primary key (ID_Furnizor);

```

Connections

MoraruMarielena

FURNIZORI

Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes Actions...

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_FURNIZOR	NUMBER(3,0)	No	(null)	1	(null)
2 NUME_FURNIZOR	VARCHAR2(20 BYTE)	Yes	(null)	2	(null)
3 ADFURNIZOR	VARCHAR2(30 BYTE)	Yes	(null)	3	(null)

### Tabela "Materiale"

```
CREATE TABLE Materiale (
    ID_Material NUMBER(3),
    ID_Furnizor NUMBER(3),
    Nume_Material VARCHAR(15),
    Pret_Unitar NUMBER(5, 2)
);
```

```
ALTER TABLE Materiale
```

```
ADD CONSTRAINT ID_MATERIAL_PK primary key (ID_Material);
```

```
ALTER TABLE Materiale
```

```
ADD CONSTRAINT ID_FURNIZOR_FK foreign key (ID_Furnizor) references Furnizori(ID_Furnizor);
```

Connections

MoraruMarielena

MATERIALE

Columns Data Model Constraints Grants Statistics Triggers Flashback Dependencies Details Partitions Indexes Actions...

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_MATERIAL	NUMBER(3,0)	No	(null)	1	(null)
2 ID_FURNIZOR	NUMBER(3,0)	Yes	(null)	2	(null)
3 NUME_MATERIAL	VARCHAR2(15 BYTE)	Yes	(null)	3	(null)
4 PRET_UNITAR	NUMBER(5,2)	Yes	(null)	4	(null)

## **2. Comenzi folosind limbajul de manipulare a datelor: INSERT, UPDATE, DELETE, MERGE.**

### **2.1. Adaugarea inregistrarilor**

- Adaugarea inregistrarilor in tabela Produse.

```
INSERT INTO Produse (ID_Produs, Nume_Produs, Tip, Pret) VALUES (1, 'Tarta vanilie', 'Dulce', 15);
```

```
INSERT INTO Produse (ID_Produs, Nume_Produs, Tip, Pret) VALUES (2, 'Tarta fistic', 'Dulce', 25);
```

```
INSERT INTO Produse (ID_Produs, Nume_Produs, Tip, Pret) VALUES (3, 'Tort visine', 'Dulce', 45);
```

```
INSERT INTO Produse (ID_Produs, Nume_Produs, Tip, Pret) VALUES (4, 'Mousse cacao', 'Dulce', 15);
```

```
INSERT INTO Produse (ID_Produs, Nume_Produs, Tip, Pret) VALUES (5, 'Cappuccino', 'Băutură', 8);
```

```
INSERT INTO Produse (ID_Produs, Nume_Produs, Tip, Pret) VALUES (6, 'Tort Tiramisu', 'Dulce', 65);
```

```
INSERT INTO Produse (ID_Produs, Nume_Produs, Tip, Pret) VALUES (7, 'Lava Cake', 'Dulce', 18);
```

```
INSERT INTO Produse (ID_Produs, Nume_Produs, Tip, Pret) VALUES (8, 'Smoothie fructe', 'Băutură', 12);
```

```
INSERT INTO Produse (ID_Produs, Nume_Produs, Tip, Pret) VALUES (9, 'Cornulet cacao', 'Dulce', 9);
```

```
INSERT INTO Produse (ID_Produs, Nume_Produs, Tip, Pret) VALUES (10, 'Clătite', 'Dulce', 22);
```

ID_PRODUS	NUME_PRODUS	TIP	PRET
1	1 Tarta vanilie	Dulce	15
2	2 Tarta fistic	Dulce	25
3	3 Tort cu visine	Dulce	45
4	4 Mousse cacao	Dulce	15
5	5 Cappuccino	Băutură	8
6	6 Tort Tiramisu	Dulce	65
7	7 Lava Cake	Dulce	18
8	8 Smoothie fructe	Băutură	12
9	9 Cornulet cacao	Dulce	9
10	10 Clătite	Dulce	22

- Adaugarea inregistrarilor in tabela Ingrediente.

```
INSERT INTO Ingrediente (ID_Ingredient, Nume_Ingredient) VALUES (1, 'Faina');
```

```
INSERT INTO Ingrediente (ID_Ingredient, Nume_Ingredient) VALUES (2, 'Zahar');
```

```
INSERT INTO Ingrediente (ID_Ingredient, Nume_Ingredient) VALUES (3, 'Oua');
```

```

INSERT INTO Ingrediente (ID_Ingredient, Nume_Ingredient) VALUES (4, 'Praf de Copt');
INSERT INTO Ingrediente (ID_Ingredient, Nume_Ingredient) VALUES (5, 'Bicarbonat');
INSERT INTO Ingrediente (ID_Ingredient, Nume_Ingredient) VALUES (6, 'Ulei');
INSERT INTO Ingrediente (ID_Ingredient, Nume_Ingredient) VALUES (7, 'Indulcitor');
INSERT INTO Ingrediente (ID_Ingredient, Nume_Ingredient) VALUES (8, 'Esenta');
INSERT INTO Ingrediente (ID_Ingredient, Nume_Ingredient) VALUES (9, 'Sare');
INSERT INTO Ingrediente (ID_Ingredient, Nume_Ingredient) VALUES (10, 'Cacao');
INSERT INTO Ingrediente (ID_Ingredient, Nume_Ingredient) VALUES (11, 'Drojdie');
INSERT INTO Ingrediente (ID_Ingredient, Nume_Ingredient) VALUES (12, 'Lapte');

```

The screenshot shows the Oracle SQL Developer interface. On the left, the Connections pane is open, showing a connection named 'MoraruMarielena'. On the right, the 'INGREDIENTE' table is displayed in the main pane. The table has two columns: 'ID\_INGREDIENT' and 'NUME\_INGREDIENT'. The data is as follows:

ID_INGREDIENT	NUME_INGREDIENT
1	Faina
2	Zahar
3	Oua
4	Praf de Copt
5	Bicarbonat
6	Ulei
7	Indulcitor
8	Esenta
9	Sare
10	Cacao
11	Drojdie
12	Lapte

- Adaugarea inregistrarilor in tabela Reteta.

```

INSERT INTO Retete (ID_Reteta, ID_Produs, ID_Ingredient, Cantitate) VALUES (1, 4, 4, 300);
INSERT INTO Retete (ID_Reteta, ID_Produs, ID_Ingredient, Cantitate) VALUES (2, 4, 6, 200);
INSERT INTO Retete (ID_Reteta, ID_Produs, ID_Ingredient, Cantitate) VALUES (3, 4, 12, 50);
INSERT INTO Retete (ID_Reteta, ID_Produs, ID_Ingredient, Cantitate) VALUES (4, 5, 5, 100);
INSERT INTO Retete (ID_Reteta, ID_Produs, ID_Ingredient, Cantitate) VALUES (5, 5, 2, 150);
INSERT INTO Retete (ID_Reteta, ID_Produs, ID_Ingredient, Cantitate) VALUES (6, 6, 6, 250);
INSERT INTO Retete (ID_Reteta, ID_Produs, ID_Ingredient, Cantitate) VALUES (7, 6, 4, 150);
INSERT INTO Retete (ID_Reteta, ID_Produs, ID_Ingredient, Cantitate) VALUES (8, 7, 7, 200);
INSERT INTO Retete (ID_Reteta, ID_Produs, ID_Ingredient, Cantitate) VALUES (9, 7, 9, 100);

```

```

INSERT INTO Retete (ID_Reteta, ID_Produs, ID_Ingredient, Cantitate) VALUES (10, 8, 8, 150);
INSERT INTO Retete (ID_Reteta, ID_Produs, ID_Ingredient, Cantitate) VALUES (11, 8, 5, 100);
INSERT INTO Retete (ID_Reteta, ID_Produs, ID_Ingredient, Cantitate) VALUES (12, 9, 10, 200);
INSERT INTO Retete (ID_Reteta, ID_Produs, ID_Ingredient, Cantitate) VALUES (13, 9, 8, 100);
INSERT INTO Retete (ID_Reteta, ID_Produs, ID_Ingredient, Cantitate) VALUES (14, 10, 11, 120);

```

The screenshot shows the Oracle SQL Developer interface. On the left, the 'Connections' sidebar is open, showing a connection named 'MoraruMarielena'. Below it is the schema browser, which lists several tables: ANGAJATI, COMENZI, FURNIZORI, INGREDIENTE, MATERIALE, PRODUSE, and RETETE. The RETETE table is selected. On the right, the main workspace displays the RETETE table with its columns: ID\_RETETA, ID\_PRODUS, ID\_INGREDIENT, and CANTITATE. The data is as follows:

ID_RETETA	ID_PRODUS	ID_INGREDIENT	CANTITATE
1	1	4	300
2	2	4	200
3	3	4	50
4	4	5	100
5	5	5	150
6	6	6	250
7	7	6	150
8	8	7	200
9	9	7	100
10	10	8	150
11	11	8	100
12	12	9	200
13	13	9	100
14	14	10	120

- Adaugarea inregistrarilor in tabela Comenzi.

```

INSERT INTO Comenzi (ID_Comanda, ID_Produs, Data_Comanda, Cantitate, Nume_Client, AdLivrare)
VALUES (1, 3, TO_DATE('2024-01-07', 'YYYY-MM-DD'), 3, 'Vasile', 'SPIndp');

```

```

INSERT INTO Comenzi (ID_Comanda, ID_Produs, Data_Comanda, Cantitate, Nume_Client, AdLivrare)
VALUES (2, 4, TO_DATE('2024-01-08', 'YYYY-MM-DD'), 1, 'Maria', 'SPUnirii');

```

```

INSERT INTO Comenzi (ID_Comanda, ID_Produs, Data_Comanda, Cantitate, Nume_Client, AdLivrare)
VALUES (3, 5, TO_DATE('2024-01-09', 'YYYY-MM-DD'), 2, 'Laur', 'MBravu');

```

```

INSERT INTO Comenzi (ID_Comanda, ID_Produs, Data_Comanda, Cantitate, Nume_Client, AdLivrare)
VALUES (4, 6, TO_DATE('2024-01-10', 'YYYY-MM-DD'), 1, 'Andrei', 'PAmzei');

```

```

INSERT INTO Comenzi (ID_Comanda, ID_Produs, Data_Comanda, Cantitate, Nume_Client, AdLivrare)
VALUES (5, 7, TO_DATE('2024-01-11', 'YYYY-MM-DD'), 4, 'Ana', 'PUnirii');

```

```

INSERT INTO Comenzi (ID_Comanda, ID_Produs, Data_Comanda, Cantitate, Nume_Client, AdLivrare)
VALUES (6, 8, TO_DATE('2024-01-12', 'YYYY-MM-DD'), 2, 'Mioara', 'CMoxa');

```

```

INSERT INTO Comenzi (ID_Comanda, ID_Produs, Data_Comanda, Cantitate, Nume_Client, AdLivrare)
VALUES (7, 9, TO_DATE('2024-01-13', 'YYYY-MM-DD'), 3, 'Narcisa', 'CBlevedere');

```

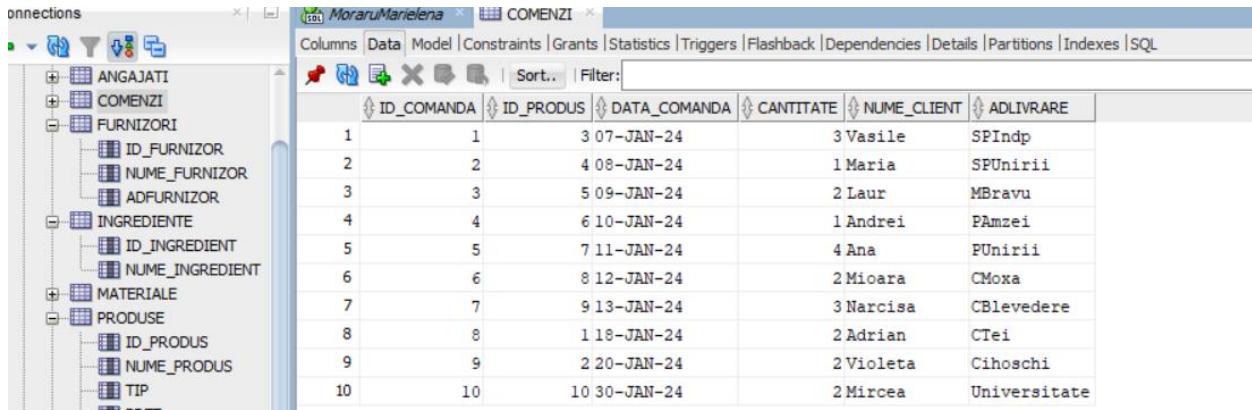
```

INSERT INTO Comenzi (ID_Comanda, ID_Produs, Data_Comanda, Cantitate, Nume_Client, AdLivrare)
VALUES (8, 1, TO_DATE('2024-01-18', 'YYYY-MM-DD'), 2, 'Adrian', 'CTei');

```

INSERT INTO Comenzi (ID\_Comanda, ID\_Produs, Data\_Comanda, Cantitate, Nume\_Client, AdLivrare)  
VALUES (9, 2, TO\_DATE('2024-01-20', 'YYYY-MM-DD'), 2, 'Violeta', 'Cihoschi');

INSERT INTO Comenzi (ID\_Comanda, ID\_Produs, Data\_Comanda, Cantitate, Nume\_Client, AdLivrare)  
VALUES (10, 10, TO\_DATE('2024-01-30', 'YYYY-MM-DD'), 2, 'Mircea', 'Universitate');



	ID_COMANDA	ID_PRODUS	DATA_COMANDA	CANTITATE	NUME_CLIENT	ADLIVRARE
1	1	3	07-JAN-24	3	Vasile	SPIndp
2	2	4	08-JAN-24	1	Maria	SPUnirii
3	3	5	09-JAN-24	2	Laur	MBravu
4	4	6	10-JAN-24	1	Andrei	PAmzei
5	5	7	11-JAN-24	4	Ana	PUnirii
6	6	8	12-JAN-24	2	Mioara	CMoxa
7	7	9	13-JAN-24	3	Narcisa	CBlevedere
8	8	1	18-JAN-24	2	Adrian	CTei
9	9	2	20-JAN-24	2	Violeta	Cihoschi
10	10	10	30-JAN-24	2	Mircea	Universitate

- Adaugarea inregistrarilor in tabela Angajati.

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (1, 'Ion', 'Alin', 'Chelner', 2600);

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (2, 'Picus', 'Laurentiu', 'Chelner', 2600);

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (3, 'Zeca', 'Bianca', 'Chelner', 2600);

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (11, 'Coros', 'Matei', 'Chelner', 2600);

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (4, 'Chirita', 'Maria', 'Barista', 3100);

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (5, 'Pricopi', 'Adrian', 'Op Casa', 2700);

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (6, 'Bunescu', 'Adina', 'CallCenter', 3000);

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (7, 'Buga', 'Andra', 'Chelner', 2500);

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (8, 'Culea', 'Darius', 'Cofetar', 6200);

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (9, 'Iorgu', 'Angelin', 'Chelner', 2800);

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (10, 'Voicu', 'Cristina', 'As.Cofetar', 3300);

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (12, 'Stan', 'Mihaela', 'SefCofetar', 8300);

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (13, 'Vilcea', 'Sorina', 'Cofetar', 3300);

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (14, 'Anastasiu', 'Alex', 'Decorator', 3300);

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (15, 'Luka', 'Andrei', 'Paza', 3300);

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (16, 'Gica', 'Narcis', 'Adminis.', 5300);

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (17, 'Petre', 'Iuliana', 'Manager', 4300);

INSERT INTO Angajati (ID\_Angajat, Nume, Prenume, Functie, Salariu) VALUES (18, 'Toma', 'Sebastian', 'Contabil', 3300);

The screenshot shows the Oracle SQL Developer interface with the 'ANGAJATI' table selected in the 'Data' tab. The table contains 18 rows of data, each representing an employee with columns: ID\_ANGAJAT, NUME, PRENUME, FUNCTIE, and SALARIU.

ID_ANGAJAT	NUME	PRENUME	FUNCTIE	SALARIU
1	1	Ion	Alin Chelner	2600
2	2	Picus	Laurentiu Chelner	2600
3	3	Zeca	Bianca Chelner	2600
4	4	Chirita	Maria Barista	3100
5	5	Pricopii	Adrian Op Casa	2700
6	6	Bunescu	Adina CallCenter	3000
7	7	Buga	Andra Chelner	2500
8	8	Culea	Darius Cofetar	6200
9	9	Iorgu	Angelin Chelner	2800
10	10	Voicu	Cristina As.Cofetar	3300
11	11	Coros	Matei Chelner	2600
12	12	Stan	Mihaela SefCofetar	8300
13	13	Vilcea	Sorina Cofetar	3300
14	14	Anastasiu	Alex Decorator	3300
15	15	Luka	Andrei Paza	3300
16	16	Gica	Narcis Adminis.	5300
17	17	Petre	Iuliana Manager	4300
18	18	Toma	Sebastian Contabil	3300

- Adaugarea inregistrarilor in tabela Furnizori.

INSERT INTO Furnizori (ID\_Furnizor, Nume\_Furnizor, AdFurnizor) VALUES (1, 'FurniCorp', 'Str. Afacerilor 123');

INSERT INTO Furnizori (ID\_Furnizor, Nume\_Furnizor, AdFurnizor) VALUES (2, 'LivraGoods', 'Str. Serviciilor 456');

INSERT INTO Furnizori (ID\_Furnizor, Nume\_Furnizor, AdFurnizor) VALUES (3, 'FoodEx', 'Str. Alimentelor 789');

INSERT INTO Furnizori (ID\_Furnizor, Nume\_Furnizor, AdFurnizor) VALUES (4, 'SweetSupplies', 'Str. Dulciurilor 101');

```

INSERT INTO Furnizori (ID_Furnizor, Nume_Furnizor, AdFurnizor) VALUES (5, 'EcoProduce', 'Str. Ecologică 112');

INSERT INTO Furnizori (ID_Furnizor, Nume_Furnizor, AdFurnizor) VALUES (6, 'UtensilWorld', 'Str. Ustensilelor 131');

INSERT INTO Furnizori (ID_Furnizor, Nume_Furnizor, AdFurnizor) VALUES (7, 'PackagingPros', 'Str. Ambalajelor 141');

INSERT INTO Furnizori (ID_Furnizor, Nume_Furnizor, AdFurnizor) VALUES (8, 'PlasticGoods', 'Str. Plasticului 152');

INSERT INTO Furnizori (ID_Furnizor, Nume_Furnizor, AdFurnizor) VALUES (9, 'FreshHarvest', 'Str. Recoltelor 165');

INSERT INTO Furnizori (ID_Furnizor, Nume_Furnizor, AdFurnizor) VALUES (10, 'TastyTreats', 'Str. Deliciilor 178');

```

The screenshot shows the Oracle SQL Developer interface. On the left, the 'Connections' sidebar is open, showing 'Oracle Connections' and 'MoraruMarielena'. Under 'Tables', the 'FURNIZORI' table is selected, showing its structure with columns: ID\_FURNIZOR, NUME\_FURNIZOR, and ADFURNIZOR. The main pane displays the data for the FURNIZORI table:

ID_FURNIZOR	NUME_FURNIZOR	ADFURNIZOR
1	1 FurniCorp	Str. Afacerilor 123
2	2 LivraGoods	Str. Serviciilor 456
3	3 FoodEx	Str. Alimentelor 789
4	4 SweetSupplies	Str. Dulciurilor 101
5	5 EcoProduce	Str. Ecologică 112
6	6 UtensilWorld	Str. Ustensilelor 131
7	7 PackagingPros	Str. Ambalajelor 141
8	8 PlasticGoods	Str. Plasticului 152
9	9 FreshHarvest	Str. Recoltelor 165
10	10 TastyTreats	Str. Deliciilor 178

- Adaugarea inregistrarilor in tabela Materiale.

```

INSERT INTO Materiale (ID_Material, ID_Furnizor, Nume_Material, Pret_Unitar) VALUES (1, 1, 'Parchet Culinar', 15.75);

INSERT INTO Materiale (ID_Material, ID_Furnizor, Nume_Material, Pret_Unitar) VALUES (2, 2, 'FolieDecorativa', 12.50);

INSERT INTO Materiale (ID_Material, ID_Furnizor, Nume_Material, Pret_Unitar) VALUES (3, 3, 'Sticle Elegante', 18.20);

INSERT INTO Materiale (ID_Material, ID_Furnizor, Nume_Material, Pret_Unitar) VALUES (4, 4, 'Paie Raffinate', 14.80);

INSERT INTO Materiale (ID_Material, ID_Furnizor, Nume_Material, Pret_Unitar) VALUES (5, 5, 'Cutii Elegante', 17.30);

```

```
INSERT INTO Materiale (ID_Material, ID_Furnizor, Nume_Material, Pret_Unitar) VALUES (6, 6, 'Furculite', 16.40);
```

```
INSERT INTO Materiale (ID_Material, ID_Furnizor, Nume_Material, Pret_Unitar) VALUES (7, 7, 'Pahare Premium', 13.90);
```

```
INSERT INTO Materiale (ID_Material, ID_Furnizor, Nume_Material, Pret_Unitar) VALUES (8, 8, 'Ambalaje Tort', 19.00);
```

```
INSERT INTO Materiale (ID_Material, ID_Furnizor, Nume_Material, Pret_Unitar) VALUES (9, 9, 'Foi de Aluminiu', 11.25);
```

```
INSERT INTO Materiale (ID_Material, ID_Furnizor, Nume_Material, Pret_Unitar) VALUES (10, 10, 'Pensule Cof', 14.00);
```

The screenshot shows the Oracle SQL Developer interface. On the left, the Connections pane shows an Oracle connection named 'MoraruMarielena' with several tables listed under 'Tables'. The 'MATERIALE' table is selected and shown in the main pane. The table has four columns: ID\_MATERIAL, ID\_FURNIZOR, NUME\_MATERIAL, and PRET\_UNITAR. The data is as follows:

ID_MATERIAL	ID_FURNIZOR	NUME_MATERIAL	PRET_UNITAR
1	1	1 Parchet Culinar	15.75
2	2	2 FolieDecorativa	12.5
3	3	3 Sticle Elegante	18.2
4	4	4 Paie Raffinate	14.8
5	5	5 Cutii Elegante	17.3
6	6	6 Furculite	16.4
7	7	7 Pahare Premium	13.9
8	8	8 Ambalaje Tort	19
9	9	9 Foi de Aluminiu	11.25
10	10	10 Pensule Cof	14

## 2.2 Actualizarea inregistrarilor

- Actualizarea salariului angajatului cu id\_angajat=2.

```
update Angajati
```

```
set Salariu=2800 where ID_Angajat=2;
```

The screenshot shows the Oracle SQL Developer interface. On the left, the Connections pane shows an Oracle connection named 'MoraruMarielena' with several tables listed under 'Tables'. The 'ANGAJATI' table is selected and shown in the main pane. The table has five columns: ID\_ANGJAT, NUME, PRENUME, FUNCTIE, and SALARIU. The data is as follows:

ID_ANGJAT	NUME	PRENUME	FUNCTIE	SALARIU
1	1 Ion	Alin	Chelner	2600
2	2 Picus	Laurentiu	Chelner	2800
3	3 Zeca	Bianca	Chelner	2600
4	4 Chirita	Maria	Barista	3100

- Actualizarea numelui produsului cu id\_produs=3.

update Produse

```
set Nume_Produs='Tort cu cirese' where ID_Produs=3;
```

ID_PR...	NUME_PRODUS	TIP	PRET
1	1 Tarta vanilie	Dulce	15
2	2 Tarta fistic	Dulce	25
3	3 Tort cu cirese	Dulce	45
4	4 Mereasa dulce	Dulce	12

- Sa se modifice data comenzii cu cantitatea mai mica decat 2.

update Comenzi

```
set Data_Commanda=TO_DATE('15-JAN-2024','DD-MM-YY') where Cantitate<2;
```

ID_COMMANDA	ID_PRODUS	DATA_COMMANDA	CANTITATE	NUME_CLIENT	ADLIVRARE
1	1	3 07-JAN-24	3	Vasile	SPIndp
2	2	4 15-JAN-24	1	Maria	SPUnirii
3	3	5 09-JAN-24	2	Laur	MBravu
4	4	6 15-JAN-24	1	Andrei	PAmzei

- Sa se modifice pretul unitar al materialului cu id\_material=9.

update Materiale

```
set Pret_Unitar=9.99 where ID_Material=9;
```

ID_MATERIAL	ID_FURNIZOR	NUME_MATERIAL	PRET_UNITAR
1	1	1 Parchet Culinar	15.75
2	2	2 FolieDecorativa	12.5
3	3	3 Sticla Elegante	18.2
4	4	4 Paie Raffinate	14.8
5	5	5 Cutii Elegante	17.3
6	6	6 Furculite	16.4
7	7	7 Pahare Premium	13.9
8	8	8 Ambalaje Tort	19
9	9	9 Foi de Aluminiu	9.99
10	10	10 Pensule Cof	14

- Sa se actualizeze adresa furnizorului cu numele ‘FoodEx’.

update Furnizori

```
set AdFurnizor='Str. Baciului 13' where Nume_Furnizor='FoodEx';
```

ID_FURNIZOR	NUME_FURNIZOR	ADFURNIZOR
1	FurniCorp	Str. Afacerilor 123
2	LivraGoods	Str. Serviciilor 456
3	FoodEx	Str. Baciului 13

## 2.3 Stergerea si recuperarea unei tabele

- Stergerea angajatului cu numele “Iorgu”.

delete from Angajati

```
where Nume='Iorgu';
```

ID_ANG...	NUME	PRENUME	FUNCTIE	SALARIU
1	Ion	Alin	Chelner	2600
2	Picus	Laurentiu	Chelner	2800
3	Zeca	Bianca	Chelner	2600
4	Chirita	Maria	Barista	3100
5	Pricopi	Adrian	Op Casa	2700
6	Bunescu	Adina	CallCenter	3000
7	Buga	Andra	Chelner	2500
8	Culea	Darius	Cofetar	6200
9	Voicu	Cristina	As.Cofetar	3300
10	Coros	Matei	Chelner	2600
11	Stan	Mihaela	SefCofetar	8300
12	Vilcea	Sorina	Cofetar	3300
13	Anastasiu	Alex	Decorator	3300
14	Luka	Andrei	Paza	3300
15	Gica	Narcis	Adminis.	5300
16	Petre	Iuliana	Manager	4300
17	Toma	Sebastian	Contabil	3300

- Stergerea tабели “Ingrediente” si recuperarea acesteia.

drop table Ingrediente cascade constraints;

flashback table Ingrediente to before drop;

The screenshot shows the Oracle SQL Developer interface. In the top window, titled 'MoraruMarielena.sql', there is a single line of SQL code:

```
drop table Ingrediente cascade constraints;
flashback table Ingrediente to before drop;
```

In the bottom window, titled 'Script Output', the results of the execution are displayed:

```
Table INGREDIENTE dropped.

Flashback succeeded.
```

The status bar at the bottom of the output window indicates: 'Task completed in 0,103 seconds'.

### **3. Exemple de interogări variate**

1. Sa se selecteze retetele ce utilizeaza ingredientul cu id\_ingredient = 4.

```
select * from Retete
```

```
where ID_Ingredient = 4
```

```
order by Cantitate;
```

The screenshot shows the Oracle SQL Developer interface. The top window is titled 'Worksheet' and contains the SQL query:

```
select * from Retete
where ID_Ingredient = 4
order by Cantitate;
```

The bottom window is titled 'Query Result' and displays the results of the query:

ID_RETETA	ID_PRODUS	ID_INGREDIENT	CANTITATE	
1	7	6	4	150
2	1	4	4	300

The results show two rows of data from the 'Retete' table where the ingredient ID is 4.

2. Sa se selecteze toate comezile care au fost comandate in cantitati de 2 dupa 12 ianuarie 2024.

```
SELECT * FROM Comenzi
```

```
WHERE Cantitate = 2
```

```
AND Data_Comanda > TO_DATE('2024-01-12', 'YYYY-MM-DD');
```

```

SELECT * FROM Comenzi
WHERE Cantitate = 2
AND Data_Commanda > TO_DATE('2024-01-12', 'YYYY-MM-DD');

```

Query Result

ID_COMANDA	ID_PRODUS	DATA_COMMANDA	CANTITATE	NUME_CLIENT	ADLIVRARE
1	8	1 18-JAN-24	2	Adrian	CTei
2	9	2 20-JAN-24	2	Violeta	Cihoschi
3	10	10 30-JAN-24	2	Mircea	Universitate

3. Sa se selecteze id-ul si numele angajatilor a căror prenume incepe cu litera A (upper functie single-row).

```

select ID_Angajat, Nume
from Angajati
where Prenume like upper('A%')
order by ID_Angajat;

```

```

select ID_Angajat, Nume
from Angajati
where Prenume like upper('A%')
order by ID_Angajat;

```

Query Result

ID_ANGAJAT	NUME
1	Ion
2	Pricopiu
3	Bunescu
4	Buga
5	Anastasiu
6	Luka

4. Sa se selecteze ID\_Furnizor, nume furnizor din tabela Furnizori si nume material din tabela Materiale si sa se realizeze jonctiunea dintre cele doua tabele.

```

SELECT d.ID_Furnizor, d.Nume_Furnizor, a.Nume_Material
from Furnizori d, Materiale a

```

where d.ID\_Furnizor = a.ID\_Furnizor;

The screenshot shows a SQL query window with the following code:

```
SELECT d.ID_Furnizor, d.Nume_Furnizor, a.Nume_Material
from Furnizori d, Materiale a
where d.ID_Furnizor = a.ID_Furnizor;
```

The results pane displays a table with 10 rows of data:

ID_FURNIZOR	NUME_FURNIZOR	NUME_MATERIAL
1	1 FurniCorp	Parchet Culinar
2	2 LivraGoods	FolieDecorativa
3	3 FoodEx	Sticle Elegante
4	4 SweetSupplies	Paie Raffinate
5	5 EcoProduce	Cutii Elegante
6	6 UtensilWorld	Furculite
7	7 PackagingPros	Pahare Premium
8	8 PlasticGoods	Ambalaje Tort
9	9 FreshHarvest	Foi de Aluminiu
10	10 TastyTreats	Pensule Cof

5. Sa se selecteze doar angajatii cu salariul mai mare de 3000 de lei.

SELECT \* from Angajati

where Salariu > 3000

order by Salariu;

The screenshot shows a SQL query window with the following code:

```
SELECT * from Angajati
where Salariu > 3000
order by Salariu;
```

The results pane displays a table with 10 rows of data:

ID_ANGAJAT	NUME	PRENUME	FUNCTIE	SALARIU
1	4 Chirita	Maria	Barista	3100
2	10 Voicu	Cristina	As.Cofetar	3300
3	14 Anastasiu	Alex	Decorator	3300
4	18 Toma	Sebastian	Contabil	3300
5	15 Luka	Andrei	Paza	3300
6	13 Vilcea	Sorina	Cofetar	3300
7	17 Petre	Iuliana	Manager	4300
8	16 Gica	Narcis	Adminis.	5300
9	8 Culea	Darius	Cofetar	6200
10	12 Stan	Mihaela	SefCofetar	8300

6. Sa se afiseze pretul minim si pretul maxim al Materialelor.

SELECT MIN(Pret\_Unitar) AS Pret\_Minim, MAX(Pret\_Unitar) AS Pret\_Maxim  
FROM Materiale;

The screenshot shows a SQL editor window titled 'Worksheet' with the following query:

```
SELECT MIN(Pret_Unitar) AS Pret_Minim, MAX(Pret_Unitar) AS Pret_Maxim
FROM Materiale;
```

Below the query, the 'Query Result' tab is selected, showing the output:

	PRET_MINIM	PRET_MAXIM
1	9.99	19

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

```
SELECT Produse.Nume_Produs, Comenzi.cantitate, Produse.Pret,
Comenzi.cantitate * Produse.Pret AS Valoare
FROM Produse, Comenzi
WHERE Produse.ID_Produs= Comenzi.ID_Produs;
```

The screenshot shows a SQL editor window titled 'Worksheet' with the following query:

```
SELECT Produse.Nume_Produs, Comenzi.cantitate, Produse.Pret,
Comenzi.cantitate * Produse.Pret AS Valoare
FROM Produse, Comenzi
WHERE Produse.ID_Produs= Comenzi.ID_Produs;
```

Below the query, the 'Query Result' tab is selected, showing the output:

NUME_PRODUS	CANTITATE	PRET	VALOARE
1 Tarta vanilie	2	15	30
2 Tarta fistic	2	25	50
3 Tort cu cirese	3	45	135
4 Mousse cacao	1	15	15
5 Cappuccino	2	8	16
6 Tort Tiramisu	1	65	65
7 Lava Cake	4	18	72
8 Smoothie fructe	2	12	24
9 Cornulet cacao	3	9	27
10 Clătite	2	22	44

8. Sa se selecteze numai produsele cu valoarea cuprinsa intre 20 si 50.

```
SELECT Produse.Nume_Produs, Comenzi.cantitate, Produse.Pret,
Comenzi.cantitate * Produse.Pret AS Valoare
```

```

FROM Produse, Comenzi
WHERE Produse.ID_Produs= Comenzi.ID_Produs
AND Comenzi.cantitate * Produse.Pret BETWEEN 20 AND 50;

```

The screenshot shows the MySQL Workbench interface with a query editor and a results grid.

```

SELECT Produse.Nume_Produs, Comenzi.cantitate, Produse.Pret,
Comenzi.cantitate * Produse.Pret AS Valoare
FROM Produse, Comenzi
WHERE Produse.ID_Produs= Comenzi.ID_Produs
AND Comenzi.cantitate * Produse.Pret BETWEEN 20 AND 50;

```

**Query Result:**

NUME_PRODUS	CANTITATE	PRET	VALOARE
1 Tarta vanilie	2	15	30
2 Tarta fistic	2	25	50
3 Smoothie fructe	2	12	24
4 Cornulet cacao	3	9	27
5 Clătite	2	22	44

9. Să se afișeze id-ul ingredientului și numele ingredientului chiar dacă nu au fost folosite.

```

SELECT i.ID_Ingredient, i.Nume_Ingredient, r.cantitate
FROM Ingrediente i, Retete r
WHERE i.ID_Ingredient = r.ID_Ingredient (+);

```

The screenshot shows the MySQL Workbench interface with a query editor and a results grid.

```

SELECT i.ID_Ingredient, i.Nume_Ingredient, r.cantitate
FROM Ingrediente i, Retete r
WHERE i.ID_Ingredient = r.ID_Ingredient (+);

```

**Query Result:**

ID_INGREDIENT	NUME_INGREDIENT	CANTITATE
1	Faina	(null)
2	Zahar	150
3	Oua	(null)
4	Praf de Copt	300
5	Praf de Copt	150
6	Bicarbonat	100
7	Bicarbonat	100
8	Ulei	200
9	Ulei	250
11	Esenta	100
12	Esenta	150
14	Cacao	200
15	Drojdie	120
16	Lapte	50

10. Sa se afiseze denumirea produsului și pretul acestuia.

```

SELECT 'Produsul: ' || INITCAP(Nume_Produs) || ' are pretul de ' || Pret || ' lei.'
FROM Produse

```

```

SELECT 'Produsul: ' || INITCAP(Nume_Produs) || ' are pretul de ' || Pret || ' lei.'
FROM Produse;

```

Query Result

	'PRODUSUL'  INITCAP(NUME_PRODUS)  'ARE PRETUL DE'  PRET  'LEI.'
1	Produsul: Tort Cu Cirese are pretul de 45 lei.
2	Produsul: Cappuccinoare pretul de 8 lei.
3	Produsul: Tort Tiramisuare pretul de 65 lei.
4	Produsul: Lava Cakeare pretul de 18 lei.
5	Produsul: Smoothie Fructeare pretul de 12 lei.
6	Produsul: Cornulet Cacaoare pretul de 9 lei.
7	Produsul: Clătitare pretul de 22 lei.
8	Produsul: Tarta Vanilieare pretul de 15 lei.
9	Produsul: Tarta Fisticare pretul de 25 lei.
10	Produsul: Mousse Cacaoare pretul de 15 lei.

11. Sa se afiseze pretul materialelor rotunjit la numar intreg.

```

SELECT Nume_Material, ROUND(Pret_Unitar) AS Pret_Rotunjit
FROM Materiale;

```

```

SELECT Nume_Material, ROUND(Pret_Unitar) AS Pret_Rotunjit
FROM Materiale;

```

Query Result

NUME_MATERIAL	PRET_ROTUNJIT
1 Parchet Culinar	16
2 Sticle Elegante	18
3 Paie Raffinate	15
4 Cutii Elegante	17
5 Pahare Premium	14
6 Foi de Aluminiu	10
7 FolieDecorativa	13
8 Furculițe	16
9 Ambalaje Tort	19
10 Pensule Cof	14

12. Să se afișeze comenzile și data comenzii și în format “MM/YY”.

```

SELECT ID_Comanda, Data_Comanda,
TO_CHAR(Data_Comanda, 'MM/YY') AS DATA2_COMANDA

```

FROM Comenzi;

The screenshot shows the Oracle SQL Developer interface. In the Worksheet tab, there is a query:

```
SELECT ID_Comanda, Data_Commanda, TO_CHAR(Data_Commanda, 'MM/YY') AS DATA2_COMMANDA
FROM Comenzi;
```

In the Query Result tab, the output is:

ID_COMANDA	DATA_COMMANDA	DATA2_COMMANDA
1	1 07-JAN-24	01/24
2	2 15-JAN-24	01/24
3	3 09-JAN-24	01/24
4	4 15-JAN-24	01/24
5	5 11-JAN-24	01/24
6	6 12-JAN-24	01/24
7	7 13-JAN-24	01/24
8	8 18-JAN-24	01/24
9	9 20-JAN-24	01/24
10	10 30-JAN-24	01/24

13. Sa se afiseze comenzile incheiate intre zilele 12 si 20 ianuarie 2024.

SELECT ID\_Commanda, Data\_Commanda, Nume\_Client, AdLivrare

FROM Comenzi

WHERE EXTRACT(MONTH FROM Data\_Commanda) = 1

AND EXTRACT(DAY FROM Data\_Commanda) BETWEEN 12 AND 20

AND EXTRACT(YEAR FROM Data\_Commanda) = 2024;

The screenshot shows the Oracle SQL Developer interface. In the Worksheet tab, there is a complex query:

```
SELECT ID_Commanda, Data_Commanda, Nume_Client, AdLivrare
FROM Comenzi
WHERE EXTRACT(MONTH FROM Data_Commanda) = 1
    AND EXTRACT(DAY FROM Data_Commanda) BETWEEN 12 AND 20
    AND EXTRACT(YEAR FROM Data_Commanda) = 2024;
```

In the Query Result tab, the output is:

ID_COMMANDA	DATA_COMMANDA	NUME_CLIENT	ADLIVRARE
1	6 12-JAN-24	Mioara	CMoxa
2	7 13-JAN-24	Narcisa	CBlevedere
3	4 15-JAN-24	Andrei	PAmzei
4	2 15-JAN-24	Maria	SFUnirii
5	8 18-JAN-24	Adrian	CTei
6	9 20-JAN-24	Violeta	Cihoschi

14. Obțineți suma totală a cantităților de produse comandate de fiecare client din tabela 'Comenzi' și afișați rezultatele grupate pe nume de client.

SELECT Nume\_Client, SUM(Cantitate) AS TotalCantitate

```
FROM Comenzi  
GROUP BY Nume_Client  
ORDER BY TotalCantitate;
```

The screenshot shows a SQL query editor interface with two tabs: 'Worksheet' and 'Query Result'. The 'Worksheet' tab contains the following SQL code:

```
SELECT Nume_Client, SUM(Cantitate) AS TotalCantitate  
FROM Comenzi  
GROUP BY Nume_Client  
ORDER BY TotalCantitate;
```

The 'Query Result' tab displays the output of the query, which is a table with two columns: 'NUME\_CLIENT' and 'TOTALCANTITATE'. The data is as follows:

NUME_CLIENT	TOTALCANTITATE
1 Andrei	1
2 Maria	1
3 Laur	2
4 Mircea	2
5 Adrian	2
6 Violeta	2
7 Mioara	2
8 Narcisa	3
9 Vasile	3
10 Ana	4

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

```
SELECT ID_Comanda, SUM(Cantitate * Pret) AS ValoareTotala  
FROM Comenzi  
JOIN Produse ON Comenzi.ID_Produs = Produse.ID_Produs  
GROUP BY ID_Comanda  
ORDER BY ValoareTotala DESC;
```

```

SELECT ID_Commanda, SUM(Cantitate * Pret) AS ValoareTotala
FROM Comenzi
JOIN Produse ON Comenzi.ID_Produs = Produse.ID_Produs
GROUP BY ID_Commanda
ORDER BY ValoareTotala DESC;

```

Query Result

ID_COMMANDA	VALOARETOTALA
1	135
2	72
3	65
4	50
5	44
6	30
7	27
8	24
9	16
10	15

16. Sa se foloseasca functia “INTERSECT” pentru intersecarea coloanei ID\_Produs din tabelele Produse si Comenzi.

SELECT ID\_Produs

FROM Comenzi

INTERSECT

SELECT ID\_Produs

FROM Produse;

```

SELECT ID_Produs
FROM Comenzi
INTERSECT
SELECT ID_Produs
FROM Produse;

```

Query Result

ID_PRODUS
3
4
5
6
7
8
9
1
2
10

17. Sa se afiseze id-ul ingredientelor care nu sunt folosite in tabela Retete.

```
SELECT ID_Ingredient
```

```
FROM Ingrediente
```

```
MINUS
```

```
SELECT ID_Ingredient
```

```
FROM Retete;
```

The screenshot shows the MySQL Workbench interface. The 'Worksheet' tab contains the SQL query:

```
SELECT ID_Ingredient
FROM Ingrediente
MINUS
SELECT ID_Ingredient
FROM Retete;
```

The 'Query Result' tab shows the output:

ID_INGREDIENT
1
2

All Rows Fetched: 2 in 0,015 seconds

18. Realizeaza uniunea intre produsele comandate si produsele ce reies din retetele cofetariei.

```
SELECT ID_Produs
```

```
FROM Comenzi
```

```
UNION
```

```
SELECT ID_Produs
```

```
FROM Retete;
```

The screenshot shows the MySQL Workbench interface. The 'Worksheet' tab contains the SQL query:

```
SELECT ID_Produs
FROM Comenzi
UNION
SELECT ID_Produs
FROM Retete;
```

The 'Query Result' tab shows the output:

ID_PRODUS
1
2
3
4
5
6
7
8
9
10

All Rows Fetched: 10 in 0,025 seconds

19. Sa se afiseze produsele ce contin in denumire cuvantul "Tarta" si "cacao".

```
SELECT Nume_Produs FROM Produse WHERE Nume_Produs like '%Tarta%'
```

UNION

```
SELECT Nume_Produs FROM Produse WHERE Nume_Produs like '%cacao%';
```

The screenshot shows the MySQL Workbench interface. In the top tab bar, there are three tabs: 'MoraruMarielena.sql' (selected), 'MoraruMarielena.sql', and 'PRODUSE'. Below the tabs is a toolbar with various icons. The main area has two tabs: 'Worksheet' and 'Query Builder'. The 'Worksheet' tab contains the SQL code:

```
SELECT Nume_Produs FROM Produse WHERE Nume_Produs like '%Tarta%'  
UNION  
SELECT Nume_Produs FROM Produse WHERE Nume_Produs like '%cacao%';
```

Below the code is a 'Query Result' tab. It shows a table with one column labeled 'NUME\_PRODUS' containing four rows:

NUME_PRODUS
1 Tarta vanilie
2 Tarta fistic
3 Cornulet cacao
4 Mousse cacao

The status bar at the bottom of the result tab indicates: 'All Rows Fetched: 4 in 0,022 seconds'.

20. Se va folosi CASE pentru a ordona produsele în funcție de tipurile "Tort" și "Smoothie".

```
SELECT Nume_Produs
```

FROM produse

```
WHERE Nume_Produs LIKE '%Tort%' OR Nume_Produs LIKE '%Smoothie%'
```

ORDER BY

CASE

```
WHEN Tip = 'Tort' THEN 1
```

```
WHEN Tip = 'Smoothie' THEN 2
```

```
ELSE 3
```

```
END, Nume_Produs;
```

```

SELECT Nume_Produs
FROM produse
WHERE Nume_Produs LIKE '%Tort%' OR Nume_Produs LIKE '%Smoothie%'
ORDER BY
CASE
    WHEN Tip = 'Tort' THEN 1
    WHEN Tip = 'Smoothie' THEN 2
    ELSE 3
END, Nume_Produs;

```

Query Result | All Rows Fetched: 3 in 0,019 seconds

NUME_PRODUS
1 Smoothie fructe
2 Tort Tiramisu
3 Tort cu cirese

21. Sa se afiseze numele ingredientelor care sunt prezente în tabela Retete.

SELECT Nume\_Ingredient

FROM Ingrediente

WHERE ID\_Ingredient IN (

SELECT ID\_Ingredient

FROM Retete

);

```

SELECT Nume_Ingredient
FROM Ingrediente
WHERE ID_Ingredient IN (
    SELECT ID_Ingredient
    FROM Retete
);

```

Query Result | All Rows Fetched: 10 in 0,027 seconds

NUME_INGREDIENT
1 Zahar
2 Praf de Copt
3 Bicarbonat
4 Ulei
5 Indulcitor
6 Esenta
7 Sare
8 Cacao
9 Drojdie
10 Lapte

22. Sa se afiseze adresele furnizorilor cu ID\_Furnizor mai mic de 5.

```
SELECT AdFurnizor  
FROM Furnizori  
WHERE ID_Furnizor < 5;
```

```
Worksheet | Query Builder  
SELECT AdFurnizor  
FROM Furnizori  
WHERE ID_Furnizor < 5;  
Query Result | All Rows Fetched: 4 in 0,023 seconds  
ADFURNIZOR  
1 Str. Afacerilor 123  
2 Str. Serviciilor 456  
3 Str. Baciului 13  
4 Str. Dulciurilor 101
```

#### **4. Gestiunea altor obiecte ale bazei de date: vederi, indecsi, sinonime, secente.**

1. Sa se realizeze o tabela virtuala cu angajatii ce au salariul de 3300 lei.

```
CREATE OR REPLACE VIEW v_Angajati_MR  
AS SELECT * FROM Angajati  
WHERE Salariu=3300;  
SELECT * FROM v_Angajati_MR;
```

```
Worksheet | Query Builder  
CREATE OR REPLACE VIEW v_Angajati_MR  
AS SELECT * FROM Angajati  
WHERE Salariu=3300;  
SELECT * FROM v_Angajati_MR;  
Script Output | Query Result | All Rows Fetched: 5 in 0,024 seconds  
ID_ANGAJAT | NUME | PRENUME | FUNCTIE | SALARIU  
1 | 10 Voicu | Cristina | As.Cofetar | 3300  
2 | 13 Vilcea | Sorina | Cofetar | 3300  
3 | 14 Anastasiu | Alex | Decorator | 3300  
4 | 15 Luka | Andrei | Paza | 3300  
5 | 18 Toma | Sebastian | Contabil | 3300
```

2. Sa se creeze un index pe tabela Retete pe coloana Cantitate.

```
CREATE INDEX idx_cantitate_retete
```

```
ON Retete (Cantitate);
```

The screenshot shows the SSMS interface with three tabs at the top: 'MoraruMarielena.sql', 'MoraruMarielena.sql', and 'RETETE'. The 'RETETE' tab is active. In the 'Worksheet' tab, the following SQL code is written:

```
CREATE INDEX idx_cantitate_retete  
ON Retete (Cantitate);
```

Below the code, the 'Script Output' pane displays the results of the query execution:

```
View V_ANGAJATI_MR created.  
>>Query Run In:Query Result  
Index IDX_CANTITATE_RETETE created.
```

3. Sterge indexul si tabela virtuala create anterior.

```
DROP VIEW v_Angajati_MR;
```

```
DROP INDEX idx_cantitate_retete;
```

The screenshot shows the SSMS interface with three tabs at the top: 'MoraruMarielena.sql', 'MoraruMarielena.sql', and 'RETETE'. The 'RETETE' tab is active. In the 'Worksheet' tab, the following SQL code is written:

```
DROP VIEW v_Angajati_MR;  
DROP INDEX idx_cantitate_retete;
```

Below the code, the 'Script Output' pane displays the results of the query execution:

```
View V_ANGAJATI_MR dropped.  
Index IDX_CANTITATE_RETETE dropped.
```

4. Sa se creeze o secventa pentru asigurarea unicitatii cheii primare din tabela Comenzi.

```
CREATE SEQUENCE seq_comenzi
```

```
START WITH 1
```

```
INCREMENT BY 1
```

```
NOCACHE
```

```
NOCYCLE;
```

The screenshot shows the Oracle SQL Developer interface. In the top tab bar, there are three tabs: 'MoraruMarielena.sql', 'MoraruMarielena.sql', and 'RETETE'. The main workspace is titled 'Worksheet' and contains the following SQL code:

```
CREATE SEQUENCE seq_comenzi
START WITH 1
INCREMENT BY 1
NOCACHE
NOCYCLE;
```

Below the workspace, the status bar indicates 'Task completed in 0,05 seconds'. The bottom pane displays the message 'Sequence SEQ\_COMENZI created.'

5. Sa se stearga secventa seq\_comenzi.

The screenshot shows the Oracle SQL Developer interface. In the top tab bar, there are three tabs: 'MoraruMarielena.sql', 'MoraruMarielena.sql', and 'RETETE'. The main workspace is titled 'Worksheet' and contains the following SQL code:

```
DROP SEQUENCE seq_comenzi;
```

Below the workspace, the status bar indicates 'Task completed in 0,057 seconds'. The bottom pane displays the message 'Sequence SEQ\_COMENZI dropped.'

6. Sa se creeze un sinonim pentru tabela Furnizori.

CREATE SYNONYM FZ for Furnizori;

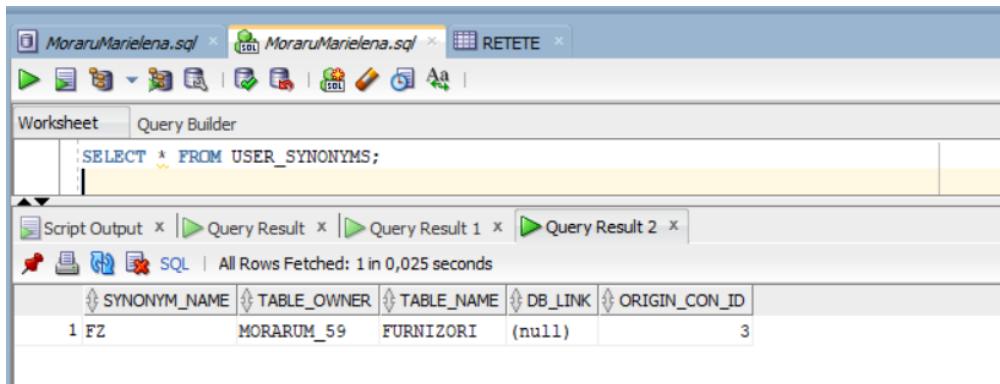
The screenshot shows the Oracle SQL Developer interface. In the top tab bar, there are three tabs: 'MoraruMarielena.sql', 'MoraruMarielena.sql', and 'RETETE'. The main workspace is titled 'Worksheet' and contains the following SQL code:

```
CREATE SYNONYM FZ for Furnizori;
```

Below the workspace, the status bar indicates 'Task completed in 0,074 seconds'. The bottom pane displays the message 'Synonym FZ created.'

7. Vizualizarea sinonimelor create.

```
SELECT * FROM USER_SYNONYMS;
```

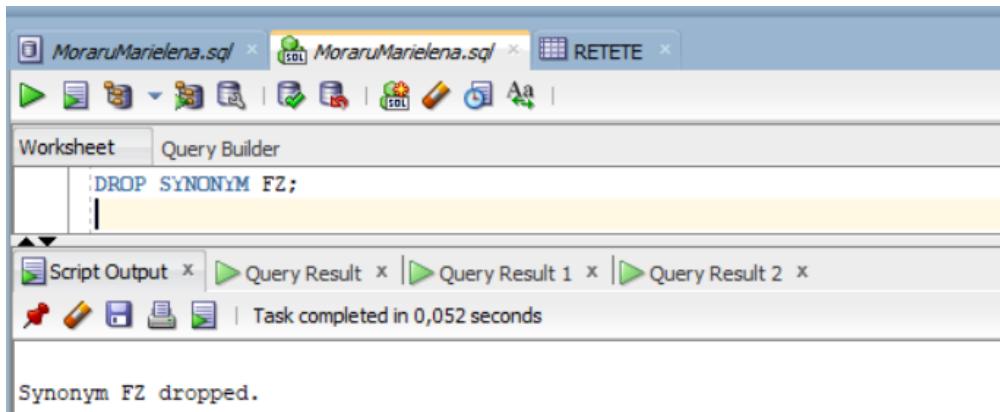


The screenshot shows the Oracle SQL Developer interface. In the top tab bar, there are three tabs: "MoraruMarielena.sql", "MoraruMarielena.sql", and "RETETE". The "Worksheet" tab is selected. In the main workspace, the query "SELECT \* FROM USER\_SYNONYMS;" is entered. Below the query, the results are displayed in a table:

SYNONYM_NAME	TABLE_OWNER	TABLE_NAME	DB_LINK	ORIGIN_CON_ID
FZ	MORARUM_59	FURNIZORI	(null)	3

8. Stergerea sinonimului FZ.

```
DROP SYNONYM FZ;
```



The screenshot shows the Oracle SQL Developer interface. In the top tab bar, there are three tabs: "MoraruMarielena.sql", "MoraruMarielena.sql", and "RETETE". The "Worksheet" tab is selected. In the main workspace, the query "DROP SYNONYM FZ;" is entered. Below the query, the message "Task completed in 0,052 seconds" is shown. At the bottom of the screen, the message "Synonym FZ dropped." is displayed.