

## **CERINTE PRIVIND REDACTAREA SI SUSTINEREA PROIECTULUI**

Proiectele grupei vor fi puse pe un singur CD si vor contine:

- cerintele proiectului,
- comenzile SQL aferente cerintelor,
- print screen-uri pentru vizualizarea rezultatelor – se va include numele utilizatorului.

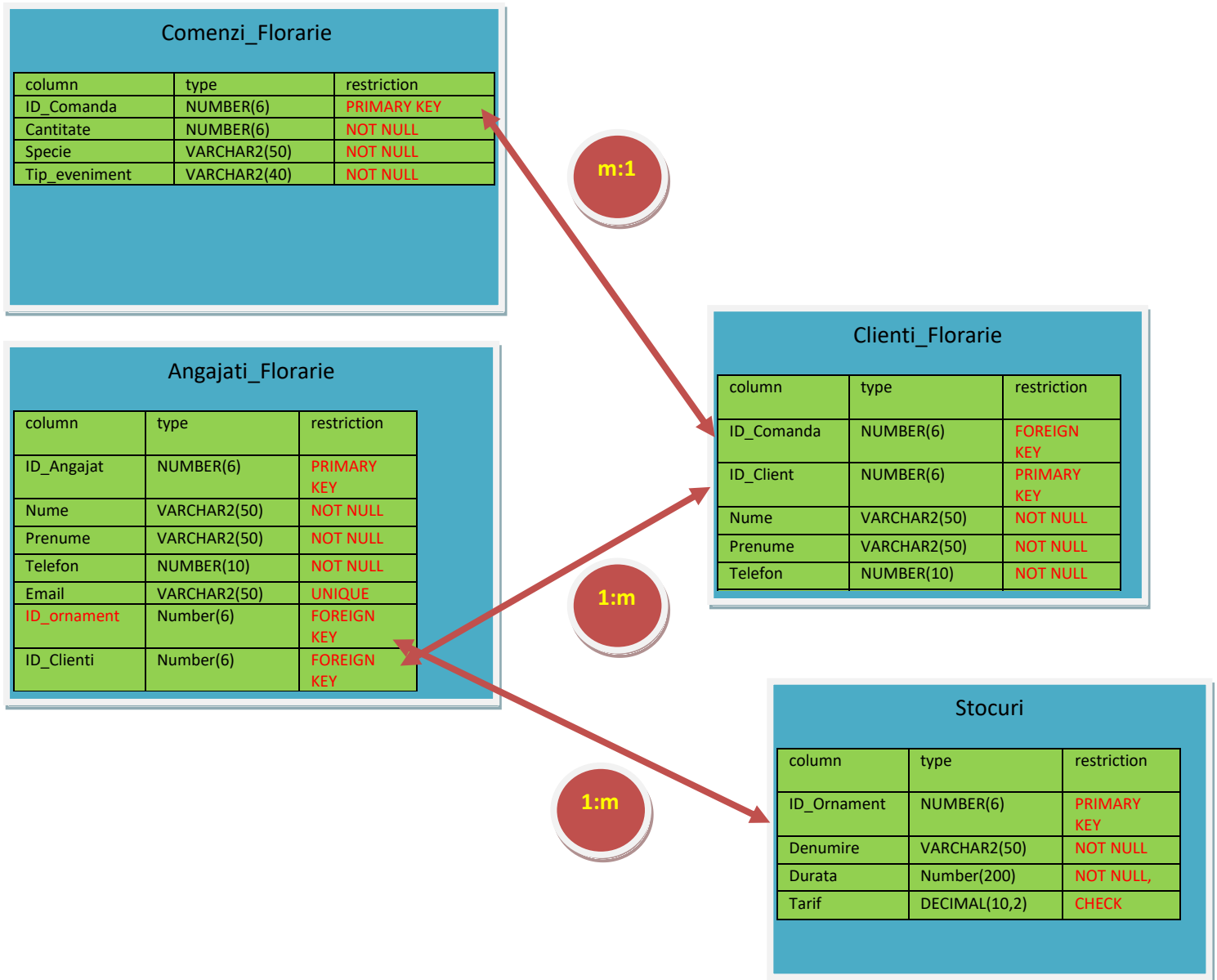
Fiecare secventa de comenzi va fi insotita de enuntul problemei (a se vedea exemplele de la seminar).

Se puncteaza:

1. Descrierea bazei de date: obiectivul proiectului, descrierea tabelelor si a atributelor, precizarea restrictiilor si a tipurilor de legaturi (max. ½ pag.). BD trebuie sa fie normalizata cel putin in FN3. (1p)
2. Schema bazei de date (1p)
3. Crearea tabelelor (min. 4 tabele) (1p)
4. Actualizarea structurii tabelelor si modificarea restrictiilor de integritate (1p)
5. Adăugarea (min 10, max 15) de înregistrări în fiecare tabelă (1p)
6. Actualizarea inregistrarilor (0,5p)
7. Stergerea si recuperarea unei tabele (0,5p)
8. Exemple de interogări variate (min 20) – inclunzând și operatorii UNION, INTERSECT, MINUS, expresiile DECODE și CASE, cereri imbricate, diverse funcții single-row, functii de grup, structuri ierarhice, jonctiuni. (1p)
9. Gestiunea altor obiecte ale bazei de date: vederi, indecsi, sinonime, secvente. (1p)

*2p din oficiu*

## Florarie-SCHEMA BD (FN3)



**Obiectivul Proiectului:** Baza de date pentru florărie are ca scop gestionarea eficientă a activităților specifice unei florării, inclusiv gestionarea stocurilor de flori și ornamente, gestionarea comenzilor de flori, precum și evidența clienților și angajaților.

**Descrierea Tabelor și Atributelor:**

#### **Tabela Comenzi\_Florarie**

ID\_Comanda (NUMBER(6), PRIMARY KEY)

ID\_Client (NUMBER(6), FOREIGN KEY)->cheie externă ce face referire la tabela "Clienti\_Florarie".

Cantitate (NUMBER(6), NOT NULL)

Specie (VARCHAR2(50), NOT NULL)

Tip\_Eveniment (VARCHAR2(40), NOT NULL)

#### **Tabela Angajati\_Florarie**

ID\_Angajat (NUMBER(6), PRIMARY KEY)

Nume (VARCHAR2(50), NOT NULL)

Prenume (VARCHAR2(50), NOT NULL)

Telefon (NUMBER(10), NOT NULL)

Email (VARCHAR2(50), UNIQUE)

#### **Tabela Clienti\_Florarie**

ID\_Client (NUMBER(6), PRIMARY KEY)

Nume (VARCHAR2(50), NOT NULL)

Prenume (VARCHAR2(50), NOT NULL)

Telefon (NUMBER(10), NOT NULL)

Email (VARCHAR2(50), UNIQUE, NOT NULL)

#### **Tabela Stocuri**

ID\_Ornament (NUMBER(6), PRIMARY KEY)

Denumire (VARCHAR2(50), NOT NULL)

Durata (NUMBER(200), NOT NULL)

Tarif (DECIMAL(10,2), CHECK)

**ID\_Angajat(NUMBER(6),FOREIGN KEY)**

Restricții:

ID\_Client în "Clienti\_Florarie" este cheie primară și trebuie să fie unic.

Email în "Angajati\_Florarie" și "Clienti\_Florarie" trebuie să fie unic.

ID\_Ornament în "Angajati\_Florarie" este cheie străină care se leagă de "Stocuri" .

ID\_Clienti în "Angajati\_Florarie" este cheie străină care se leagă de "Clienti\_Florarie".

**Explicații leături :**

**Comenzi\_Florarie și Clienti\_Florarie:**

Mai multe comenzi pot fi plasate de un client , o comandă poate avea un singur client care a plasat-o. Este o relație de Mulți la Unu , cheia de legătură este ID\_Comanda din Clienti\_Florarie care face referire la ID\_Comanda din Comenzi\_Florarie.

### Angajati\_Florarie și Clienti\_Florarie:

Un angajat poate fi asociat cu mai mulți clienți, dar un client este asociat cu un singur angajat. Este o relație de

Unul la Multi, cheia de legătură este ID\_Clienti din Angajati\_Florarie care face referire la ID\_Client din Clienti\_Florarie.

### Angajati\_Florarie și Stocuri:

Un angajat poate realiza mai multe ornamente, iar un ornament din stoc este realizat de un singur angajat. Este o relație de Unu la Mulți, cheia de legătură este ID\_Ornament din Stocuri care face referire la ID\_Ornament din Angajati\_Florarie.

## Crearea Tabelor

```
CREATE TABLE Comenzi_Florarie (  
  
    ID_Comanda NUMBER(6) PRIMARY KEY,  
  
    Cantitate NUMBER(6) NOT NULL,  
  
    Specie VARCHAR2(50) NOT NULL,  
  
    Tip_Eveniment VARCHAR2(40) NOT NULL  
  
);
```

Desc Comenzi\_Florarie;

The screenshot shows the Oracle APEX SQL Workshop interface. The top pane displays the SQL command to create the 'Comenzi\_Florarie' table. The bottom pane shows the 'Describe' view of the table, which lists its columns, data types, lengths, and constraints.

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
COMENZI_FLORARIE	ID_COMANDA	NUMBER	6		0	1	-	-	-
	CANTITATE	NUMBER	6		0	-	-	-	-
	SPECIE	VARCHAR2	50	-	-	-	-	-	-
	TIP_EVENIMENT	VARCHAR2	40	-	-	-	-	-	-

```
CREATE TABLE Clienti_Florarie (  
  
    ID_Comanda NUMBER(6) REFERENCES Comenzi_Florarie(ID_Comanda),  
  
    ID_Client NUMBER(6) PRIMARY KEY,  
  
    Nume VARCHAR2(50) NOT NULL,  
  
    Prenume VARCHAR2(50) NOT NULL,  
  
    Telefon NUMBER(10) NOT NULL
```

);

Desc Clienti\_Florarie;

The screenshot shows the APEX SQL Workshop interface. The top bar includes the APEX logo and navigation tabs: App Builder, SQL Workshop (selected), Team Development, and Gallery. A search bar and user profile (Mangiru Georgiana) are on the right. The main area is titled 'SQL Commands' and contains the following SQL code:

```
1 CREATE TABLE Clienti_Florarie (  
2   ID_Comanda NUMBER(6) REFERENCES Comenzi_Florarie(ID_Comanda),  
3   ID_Client NUMBER(6) PRIMARY KEY,  
4   Nume VARCHAR2(50) NOT NULL,  
5   Prenume VARCHAR2(50) NOT NULL,  
6   Telefon NUMBER(10) NOT NULL  
7 );  
8 Desc Clienti_Florarie;
```

Below the code editor, the 'Describe' tab is active, showing the table structure for 'CLIENTI\_FLORARIE'.

Object Type	TABLE	Object	CLIENTI_FLORARIE						
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CLIENTI_FLORARIE	ID_COMANDA	NUMBER	-	6	0	-	-	-	-
	ID_CLIENT	NUMBER	-	6	0	1	-	-	-
	NUME	VARCHAR2	50	-	-	-	-	-	-
	PRENUME	VARCHAR2	50	-	-	-	-	-	-
	TELEFON	NUMBER	-	10	0	-	-	-	-

CREATE TABLE Stocuri (

ID\_Ornament NUMBER(6) PRIMARY KEY,

Denumire VARCHAR2(50) NOT NULL,

Durata NUMBER(10) NOT NULL,

Tarif DECIMAL(10,2) CHECK (Tarif >= 0.0)

);

Desc Stocuri;

The screenshot shows the APEX SQL Workshop interface. The top bar includes the APEX logo and navigation tabs: App Builder, SQL Workshop (selected), Team Development, and Gallery. A search bar and user profile (Mangiru Georgiana) are on the right. The main area is titled 'SQL Commands' and contains the following SQL code:

```
1 CREATE TABLE Stocuri (  
2   ID_Ornament NUMBER(6) PRIMARY KEY,  
3   Denumire VARCHAR2(50) NOT NULL,  
4   Durata NUMBER(10) NOT NULL,  
5   Tarif DECIMAL(10,2) CHECK (Tarif >= 0.0)  
6 );  
7 Desc Stocuri;
```

Below the code editor, the 'Describe' tab is active, showing the table structure for 'STOCURI'.

Object Type	TABLE	Object	STOCURI						
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
STOCURI	ID_ORNAMENT	NUMBER	-	6	0	1	-	-	-
	DENUMIRE	VARCHAR2	50	-	-	-	-	-	-
	DURATA	NUMBER	-	10	0	-	-	-	-
	TARIF	NUMBER	-	10	2	-	-	-	-

CREATE TABLE Angajati\_Florarie (

ID\_Angajat NUMBER(6) PRIMARY KEY,

Nume VARCHAR2(50) NOT NULL,

Prenume VARCHAR2(50) NOT NULL,

Telefon NUMBER(10) NOT NULL,

Email VARCHAR2(50) UNIQUE,

ID\_Clienti NUMBER(6) REFERENCES Clienti\_Florarie(ID\_Client)

ID\_Ornament NUMBER(6) REFERENCES Stocuri(ID\_Stocuri)

);

Desc Angajati\_Florarie;

The screenshot shows the APEX SQL Workshop interface. The top bar includes the APEX logo, navigation tabs (App Builder, SQL Workshop, Team Development, Gallery), a search bar, and a user profile for 'Mingru Georgiana'. The main area is titled 'SQL Commands' and contains a text editor with the following SQL code:

```
1 CREATE TABLE Angajati_Florarie (  
2   ID_Angajat NUMBER(6) PRIMARY KEY,  
3   Nume VARCHAR2(50) NOT NULL,  
4   Prenume VARCHAR2(50) NOT NULL,  
5   Telefon NUMBER(10) NOT NULL,  
6   Email VARCHAR2(50) UNIQUE,  
7   ID_Ornament NUMBER(6) REFERENCES Stocuri(ID_Ornament),  
8   ID_Clienti NUMBER(6) REFERENCES Clienti_Florarie(ID_Client)  
9 );  
10 DESC Angajati_Florarie;
```

Below the editor, the 'Results' tab is active, displaying the table structure for 'ANGAJATI\_FLORARIE'. The table has 8 columns: ID\_Angajat, Nume, Prenume, Telefon, Email, ID\_Ornament, ID\_Clienti, and a primary key constraint on ID\_Angajat.

Object Type	TABLE	Object	ANGAJATI_FLORARIE						
Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ANGAJATI_FLORARIE	ID_Angajat	NUMBER	-	6	0	1	-	-	-
	Nume	VARCHAR2	50	-	-	-	-	-	-
	Prenume	VARCHAR2	50	-	-	-	-	-	-
	Telefon	NUMBER	-	10	0	-	-	-	-
	Email	VARCHAR2	50	-	-	-	✓	-	-
	ID_Ornament	NUMBER	-	6	0	-	✓	-	-
	ID_Clienti	NUMBER	-	6	0	-	✓	-	-

## Actualizarea structurii tabelelor si modificarea restrictiilor de integritate

1. Sa se modifice tipul de date al coloanei email.

**ALTER TABLE Angajati\_Florarie  
MODIFY (email VARCHAR2(30));  
Desc Angajati\_Florarie;**

APEX App Builder SQL Workshop Team Development Gallery

SQL Commands

Language: SQL Rows: 10 Clear Command Find Tables

```

1 ALTER TABLE Angajati_Florarie
2 MODIFY (email VARCHAR2(30));
3 Desc Angajati_Florarie;

```

Results Explain Describe Saved SQL History

Object Type	TABLE	Object	ANGAJATI_FLORARIE
Table	Column	Data Type	Length Precision Scale Primary Key Nullable Default Comment
ANGAJATI_FLORARIE	ID_ANGAJAT	NUMBER	- 8 0 1 - - -
	NUME	VARCHAR2	50 - - - - -
	PRENUME	VARCHAR2	50 - - - - -
	TELEFON	NUMBER	- 10 0 - - - -
	EMAIL	VARCHAR2	30 - - - - -
	ID_ORNAMENT	NUMBER	- 6 0 - - - -
	ID_CLIENTI	NUMBER	- 6 0 - - - -

2. Sa se adauge coloana varsta in tabela Angajati\_Florarie

**ALTER TABLE Angajati\_Florarie**

**ADD (varsta NUMBER(2));**

**Desc Angajati\_Florarie;**

APEX App Builder SQL Workshop Team Development Gallery

SQL Commands

Language: SQL Rows: 10 Clear Command Find Tables

```

1 ALTER TABLE Angajati_Florarie
2 ADD (varsta NUMBER(2));
3 Desc Angajati_Florarie;

```

Results Explain Describe Saved SQL History

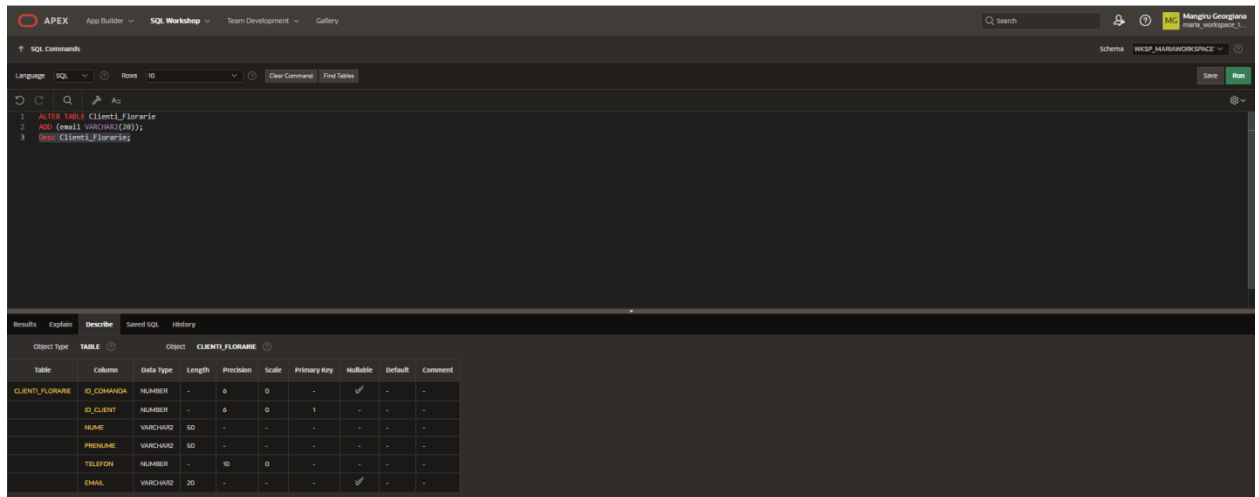
Object Type	TABLE	Object	ANGAJATI_FLORARIE
Table	Column	Data Type	Length Precision Scale Primary Key Nullable Default Comment
ANGAJATI_FLORARIE	ID_ANGAJAT	NUMBER	- 8 0 1 - - -
	NUME	VARCHAR2	50 - - - - -
	PRENUME	VARCHAR2	50 - - - - -
	TELEFON	NUMBER	- 10 0 - - - -
	EMAIL	VARCHAR2	30 - - - - -
	ID_ORNAMENT	NUMBER	- 6 0 - - - -
	ID_CLIENTI	NUMBER	- 6 0 - - - -
	VARSTA	NUMBER	- 2 0 - - - -

3. Sa se adauge coloana email in tabela Clienti

**ALTER TABLE Clienti\_Florarie**

**ADD (email VARCHAR2(20));**

**Desc Clienti\_Florarie;**

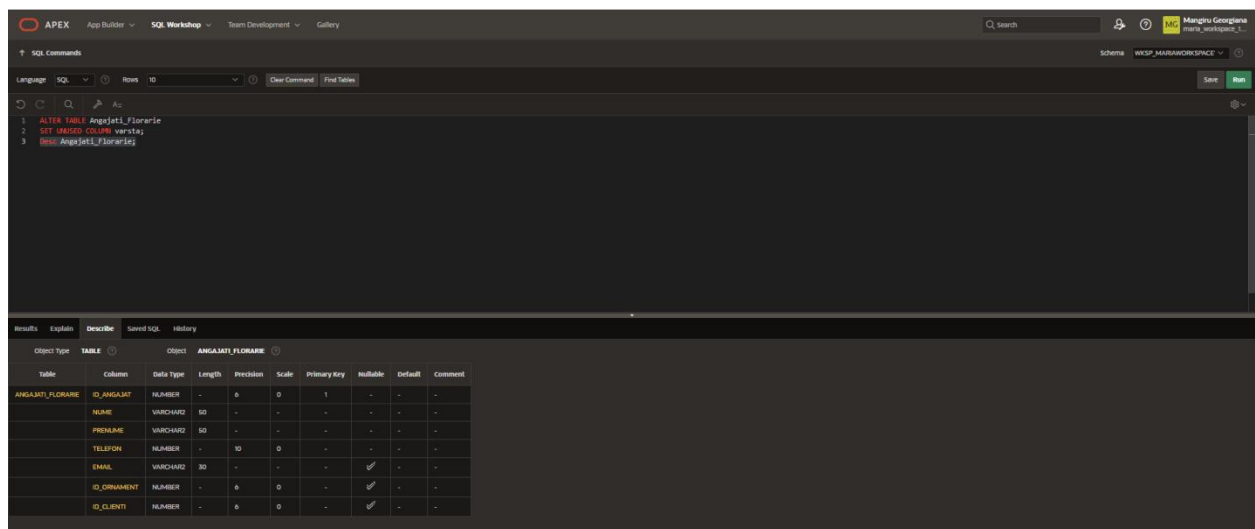


4. Sa se inactiveze coloana varsta a tabelii Angajati\_Florarie

**ALTER TABLE Angajati\_Florarie**

**SET UNUSED COLUMN varsta;**

**Desc Angajati\_Florarie;**

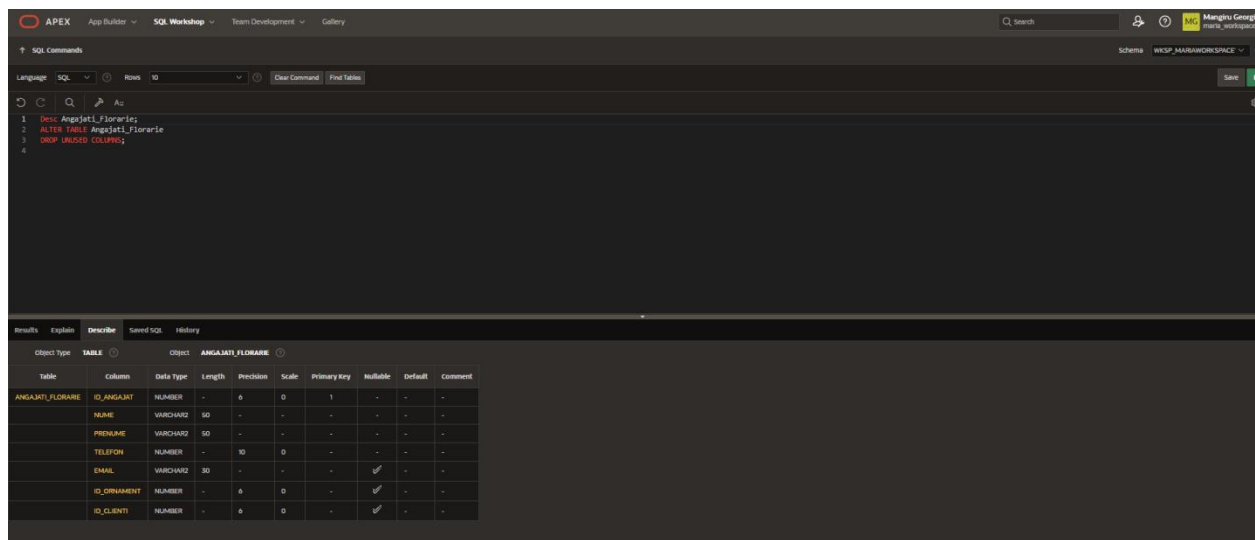


5. Sa se stearga coloanele inactive

**ALTER TABLE Angajati\_Florarie**

**DROP UNUSED COLUMNS;**

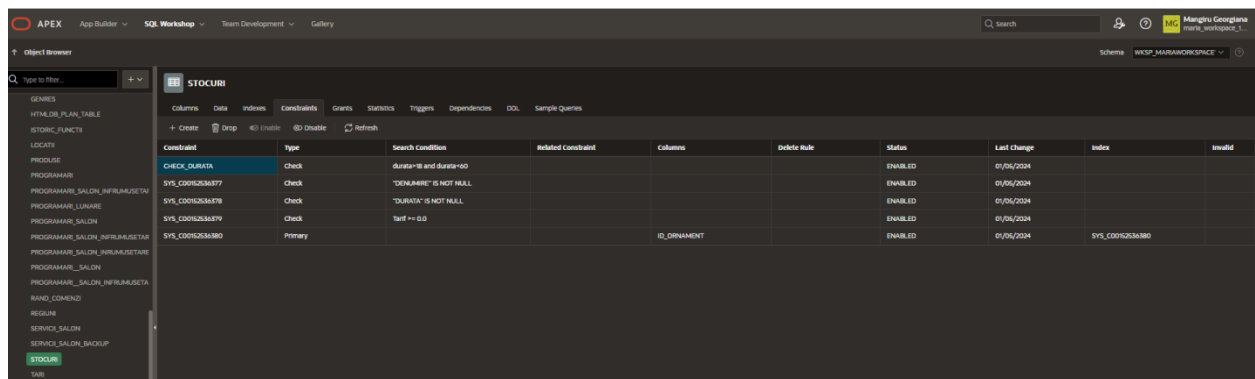




6. Sa se adauge o restrictie pe coloana durata

**ALTER TABLE Stocuri**

**ADD CONSTRAINT check\_durata CHECK (durata>18 and durata<60);**



Adăugarea (min 10, max 15) de înregistrări în fiecare tabelă

Tabela Stocuri:

Desc Stocuri;

INSERT INTO STOCURI (ID\_ORNAMENT, DENUMIRE, DURATA, TARIF) VALUES (1, 'Ornament1', 22, 15.99);

INSERT INTO STOCURI (ID\_ORNAMENT, DENUMIRE, DURATA, TARIF) VALUES (2, 'Ornament2', 40, 22.99);

INSERT INTO STOCURI (ID\_ORNAMENT, DENUMIRE, DURATA, TARIF) VALUES (3, 'Ornament3', 22, 17.50);

INSERT INTO STOCURI (ID\_ORNAMENT, DENUMIRE, DURATA, TARIF) VALUES (4, 'Ornament4', 35, 25.00);

INSERT INTO STOCURI (ID\_ORNAMENT, DENUMIRE, DURATA, TARIF) VALUES (5, 'Ornament5', 28, 20.50);

INSERT INTO STOCURI (ID\_ORNAMENT, DENUMIRE, DURATA, TARIF) VALUES (6, 'Ornament6', 45, 30.00);

INSERT INTO STOCURI (ID\_ORNAMENT, DENUMIRE, DURATA, TARIF) VALUES (7, 'Ornament7', 50, 40.00);

INSERT INTO STOCURI (ID\_ORNAMENT, DENUMIRE, DURATA, TARIF) VALUES (8, 'Ornament8', 18, 15.00);

INSERT INTO STOCURI (ID\_ORNAMENT, DENUMIRE, DURATA, TARIF) VALUES (9, 'Ornament9', 32, 23.50);

INSERT INTO STOCURI (ID\_ORNAMENT, DENUMIRE, DURATA, TARIF) VALUES (10, 'Ornament10', 19, 16.00);

INSERT INTO STOCURI (ID\_ORNAMENT, DENUMIRE, DURATA, TARIF) VALUES (11, 'Ornament11', 45, 35.00);

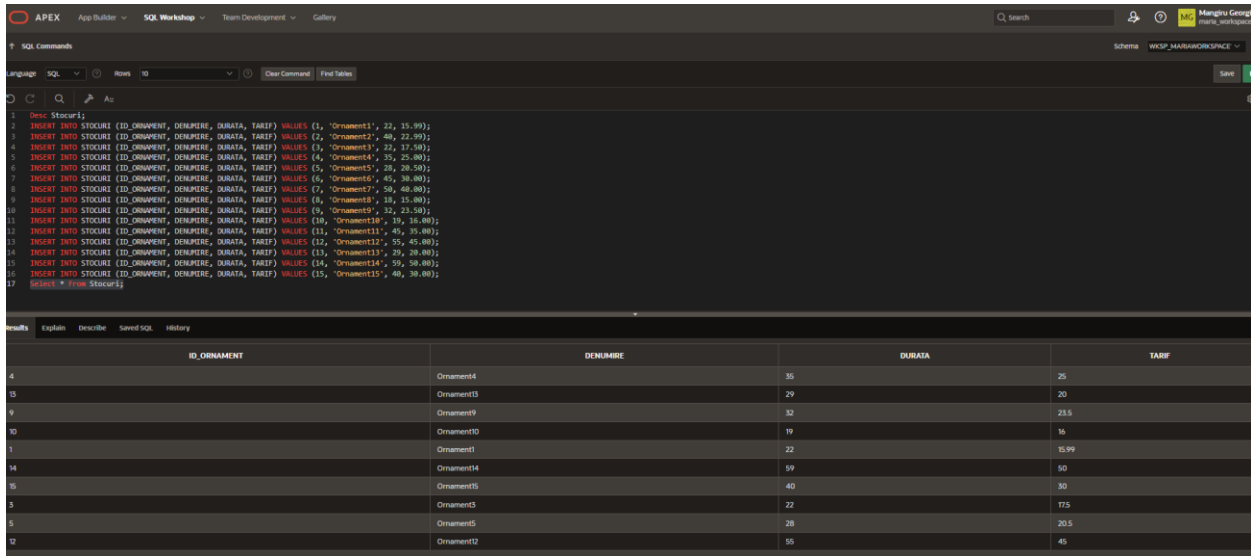
INSERT INTO STOCURI (ID\_ORNAMENT, DENUMIRE, DURATA, TARIF) VALUES (12, 'Ornament12', 55, 45.00);

INSERT INTO STOCURI (ID\_ORNAMENT, DENUMIRE, DURATA, TARIF) VALUES (13, 'Ornament13', 29, 20.00);

INSERT INTO STOCURI (ID\_ORNAMENT, DENUMIRE, DURATA, TARIF) VALUES (14, 'Ornament14', 59, 50.00);

INSERT INTO STOCURI (ID\_ORNAMENT, DENUMIRE, DURATA, TARIF) VALUES (15, 'Ornament15', 40, 30.00);

Select \* From Stocuri;



The screenshot shows the APEX SQL Workshop interface. The top bar includes the APEX logo, navigation tabs (App Builder, SQL Workshop, Team Development, Gallery), a search bar, and a user profile for 'Mangiu Georg'. The main area is titled 'SQL Commands' and contains a list of 15 INSERT statements for the STOCURI table, followed by a 'Select \* From Stocuri;' query. Below the commands, the 'Results' tab is active, displaying a table with 5 columns: ID\_ORNAMENT, DENUMIRE, DURATA, and TARIF. The table contains 15 rows of data corresponding to the inserted records.

ID_ORNAMENT	DENUMIRE	DURATA	TARIF
4	Ornament4	35	25
13	Ornament13	29	20
9	Ornament9	32	23.5
10	Ornament10	19	16
1	Ornament1	22	15.99
14	Ornament14	59	50
15	Ornament15	40	30
5	Ornament5	28	17.5
5	Ornament5	28	20.5
12	Ornament12	55	45

## Tabela Comenzi\_Florarie

Desc Comenzi\_Florarie;

INSERT INTO COMENZI\_FLORARIE (ID\_COMANDA, CANTITATE, SPECIE, TIP\_EVENIMENT) VALUES (1, 10, 'Trandafiri', 'Nunta');

INSERT INTO COMENZI\_FLORARIE (ID\_COMANDA, CANTITATE, SPECIE, TIP\_EVENIMENT) VALUES (2, 15, 'Lalele', 'Botez');

INSERT INTO COMENZI\_FLORARIE (ID\_COMANDA, CANTITATE, SPECIE, TIP\_EVENIMENT) VALUES (3, 7, 'Orhidee alba', 'Botez');

INSERT INTO COMENZI\_FLORARIE (ID\_COMANDA, CANTITATE, SPECIE, TIP\_EVENIMENT) VALUES (4, 4, 'Bujori roz', 'Nunta');

INSERT INTO COMENZI\_FLORARIE (ID\_COMANDA, CANTITATE, SPECIE, TIP\_EVENIMENT) VALUES (5, 6, 'Garoafe rosii', 'Aniversare');

```
INSERT INTO COMENZI_FLORARIE (ID_COMANDA, CANTITATE, SPECIE, TIP_EVENTIMENT) VALUES (6, 8, 'Crini albi', 'Botez');

INSERT INTO COMENZI_FLORARIE (ID_COMANDA, CANTITATE, SPECIE, TIP_EVENTIMENT) VALUES (7, 2, 'Trandafiri albi', 'Nunta');

INSERT INTO COMENZI_FLORARIE (ID_COMANDA, CANTITATE, SPECIE, TIP_EVENTIMENT) VALUES (8, 5, 'Lalele roz', 'Aniversare');

INSERT INTO COMENZI_FLORARIE (ID_COMANDA, CANTITATE, SPECIE, TIP_EVENTIMENT) VALUES (9, 3, 'Orhidee roz', 'Botez');

INSERT INTO COMENZI_FLORARIE (ID_COMANDA, CANTITATE, SPECIE, TIP_EVENTIMENT) VALUES (10, 6, 'Bujori albi', 'Nunta');

INSERT INTO COMENZI_FLORARIE (ID_COMANDA, CANTITATE, SPECIE, TIP_EVENTIMENT) VALUES (11, 4, 'Garoafe galbene', 'Aniversare');

INSERT INTO COMENZI_FLORARIE (ID_COMANDA, CANTITATE, SPECIE, TIP_EVENTIMENT) VALUES (12, 7, 'Crini roz', 'Botez');

INSERT INTO COMENZI_FLORARIE (ID_COMANDA, CANTITATE, SPECIE, TIP_EVENTIMENT) VALUES (13, 5, 'Trandafiri galbeni', 'Nunta');

INSERT INTO COMENZI_FLORARIE (ID_COMANDA, CANTITATE, SPECIE, TIP_EVENTIMENT) VALUES (14, 3, 'Lalele albe', 'Aniversare');

Select * From Comenzi_Florarie;
```

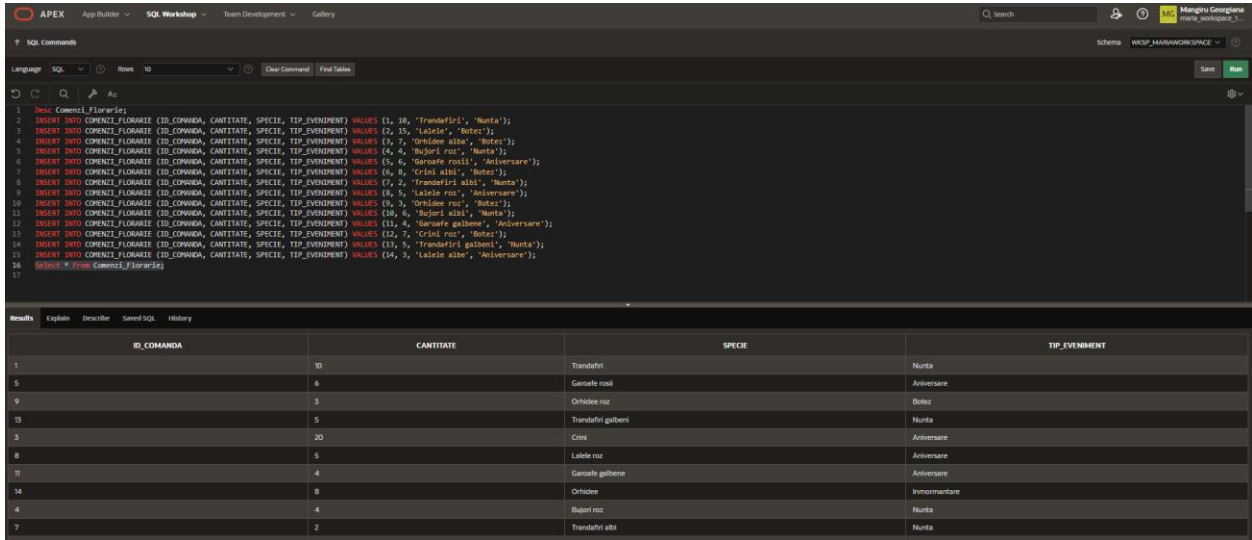


Tabela Clienti\_Florarie

```
Desc Clienti_Florarie;

INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)

VALUES (1, 1, 'Popescu', 'Ana', '0712345678', 'popescu@example.com');

INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)

VALUES (2, 2, 'Ionescu', 'Mihai', '0723456789', 'ionescu@example.com');

INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)

VALUES (3, 3, 'Georgescu', 'Elena', '0734567890', 'georgesc@example.com');

INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)
```

```
VALUES (4, 4, 'Vasilescu', 'Ion', '0745678901', 'vasilesc@example.com');

INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)

VALUES (5, 5, 'Dumitrescu', 'Ana', '0756789012', 'dumitre@example.com');

INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)

VALUES (6, 6, 'Stoica', 'Mirela', '0767890123', 'stoica@example.com');

INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)

VALUES (7, 7, 'Gheorghiu', 'Adrian', '0778901234', 'gheorghii@example.com');

INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)

VALUES (8, 8, 'Radulescu', 'Elena', '0789012345', 'radulesc@example.com');

INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)

VALUES (9, 9, 'Alexandrescu', 'George', '0790123456', 'alexandr@example.com');

INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)

VALUES (10, 10, 'Cristea', 'Maria', '0801234567', 'cristea@example.com');

INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)

VALUES (11, 11, 'Popa', 'Victor', '0812345678', 'vicpopa@example.com');

INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)

VALUES (12, 12, 'Constantinescu', 'Raluca', '0823456789', 'constant@example.com');

INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)

VALUES (13, 13, 'Iordache', 'Silviu', '0834567890', 'iordache@example.com');

INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)

VALUES (14, 14, 'Nistor', 'Alina', '0845678901', 'nistor@example.com');

INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)

VALUES (15, 15, 'Dinu', 'Daniel', '085678901', 'daniel@example.com');

Select* from Clienti_Florarie;
```

SQL Commands

```

21 VALUES (18, 18, 'Cristea', 'Maria', '0801234567', 'cristea@example.com');
22 INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)
23 VALUES (11, 11, 'Popa', 'Victor', '0612345678', 'vicpopa@example.com');
24 INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)
25 VALUES (12, 12, 'Constantinescu', 'Raluca', '0612345678', 'constant@example.com');
26 INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)
27 VALUES (13, 13, 'Tomache', 'Steluta', '0634567890', 'stelutache@example.com');
28 INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)
29 VALUES (14, 14, 'Nistor', 'Alina', '0656789012', 'alinanistor@example.com');
30 INSERT INTO CLIENTI_FLORARIE (ID_COMANDA, ID_CLIENT, NUME, PRENUME, TELEFON, EMAIL)
31 VALUES (15, 15, 'Dinu', 'Daniel', '0656789012', 'daniel@example.com');
32 SELECT * FROM CLIENTI_FLORARIE;

```

Results

ID_COMANDA	ID_CLIENT	NUME	PRENUME	TELEFON	EMAIL
5	3	Georgescu	Elena	75467890	georgescu@example.com
5	5	Dumitrescu	Ana	75678901	dumitrescu@example.com
2	2	Ionescu	Mihai	723456789	ionescu@example.com
12	12	Constantinescu	Raluca	823456789	constant@example.com
1	1	Popescu	Ana	72345678	popescu@example.com
4	4	Vasilescu	Ion	745678901	vasilescu@example.com
7	7	Gheorghiu	Adrian	778901234	gheorghiu@example.com
9	9	Alexandrescu	George	790123456	alexandrescu@example.com
13	13	Iordache	Silvia	854567890	iordache@example.com
14	14	Nistor	Alina	845678901	nistor@example.com

## Tabela Angajati\_Florarie:

Desc Angajati\_Florarie;

INSERT INTO ANGAJATI\_FLORARIE (ID\_ANGAJAT, NUME, PRENUME, TELEFON, EMAIL, ID\_ORNAMENT, ID\_CLIENTI)VALUES (16, 'Popa', 'Andreea', '0731122334', 'andreea.popa@example.com', 3, 5);

INSERT INTO ANGAJATI\_FLORARIE (ID\_ANGAJAT, NUME, PRENUME, TELEFON, EMAIL, ID\_ORNAMENT, ID\_CLIENTI) VALUES (2, 'Popescu', 'Ana', '0712345678', 'ana.popescu@example.com', 1, 1);

INSERT INTO ANGAJATI\_FLORARIE (ID\_ANGAJAT, NUME, PRENUME, TELEFON, EMAIL, ID\_ORNAMENT, ID\_CLIENTI) VALUES (3, 'Ionescu', 'Mihai', '0723456789', 'mihai.ionescu@example.com', 2, 2);

INSERT INTO ANGAJATI\_FLORARIE (ID\_ANGAJAT, NUME, PRENUME, TELEFON, EMAIL, ID\_ORNAMENT, ID\_CLIENTI) VALUES (4, 'Georgescu', 'Elena', '0734567890', 'elena.georgescu@example.com', 3, 3);

INSERT INTO ANGAJATI\_FLORARIE (ID\_ANGAJAT, NUME, PRENUME, TELEFON, EMAIL, ID\_ORNAMENT, ID\_CLIENTI) VALUES (5, 'Radu', 'Andrei', '0755555666', 'andrei.radu@example.com', 4, 4);

INSERT INTO ANGAJATI\_FLORARIE (ID\_ANGAJAT, NUME, PRENUME, TELEFON, EMAIL, ID\_ORNAMENT, ID\_CLIENTI) VALUES (6, 'Dumitru', 'Ionut', '0766666777', 'ionut.dumitru@example.com', 5, 5);

INSERT INTO ANGAJATI\_FLORARIE (ID\_ANGAJAT, NUME, PRENUME, TELEFON, EMAIL, ID\_ORNAMENT, ID\_CLIENTI) VALUES (7, 'Constantinescu', 'Gabriela', '0777777888', 'constantinescu@example.com', 6, 6);

INSERT INTO ANGAJATI\_FLORARIE (ID\_ANGAJAT, NUME, PRENUME, TELEFON, EMAIL, ID\_ORNAMENT, ID\_CLIENTI) VALUES (8, 'Iancu', 'Cristina', '0788888999', 'cristina.iancu@example.com', 7, 7);

INSERT INTO ANGAJATI\_FLORARIE (ID\_ANGAJAT, NUME, PRENUME, TELEFON, EMAIL, ID\_ORNAMENT, ID\_CLIENTI) VALUES (9, 'Moldovan', 'Vlad', '0799999000', 'moldovan@example.com', 8, 8);

INSERT INTO ANGAJATI\_FLORARIE (ID\_ANGAJAT, NUME, PRENUME, TELEFON, EMAIL, ID\_ORNAMENT, ID\_CLIENTI) VALUES (10, 'Cristea', 'Andreea', '0800000111', 'andreea.cristea@example.com', 9, 9);

INSERT INTO ANGAJATI\_FLORARIE (ID\_ANGAJAT, NUME, PRENUME, TELEFON, EMAIL, ID\_ORNAMENT, ID\_CLIENTI) VALUES (11, 'Radulescu', 'Marius', '0822222333', 'marius.radulescu@example.com', 11, 11);

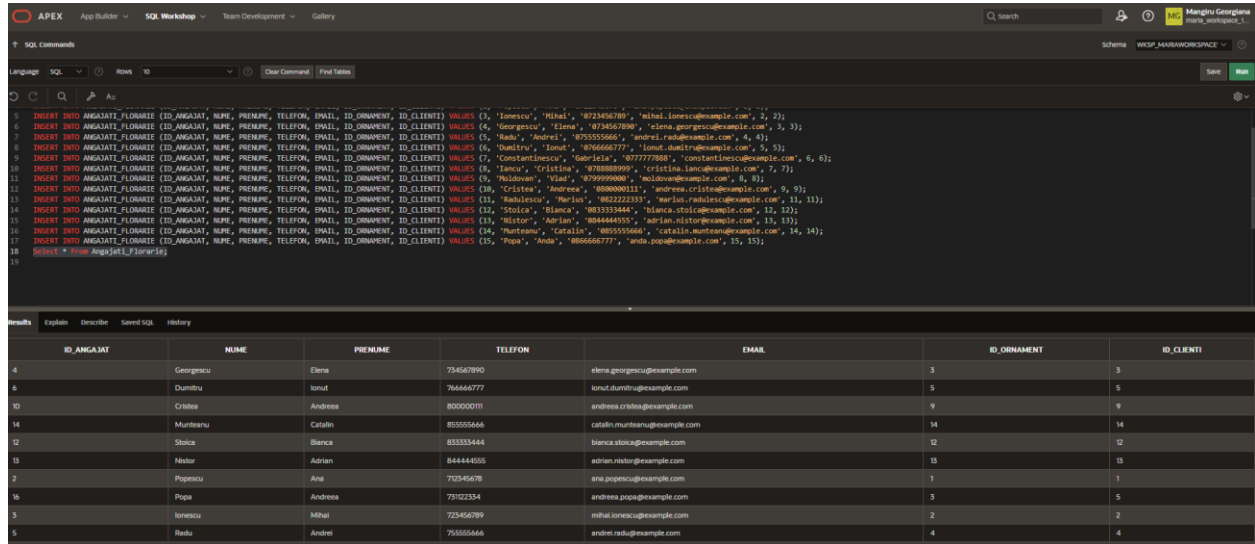
INSERT INTO ANGAJATI\_FLORARIE (ID\_ANGAJAT, NUME, PRENUME, TELEFON, EMAIL, ID\_ORNAMENT, ID\_CLIENTI) VALUES (12, 'Stoica', 'Bianca', '0833333444', 'bianca.stoica@example.com', 12, 12);

INSERT INTO ANGAJATI\_FLORARIE (ID\_ANGAJAT, NUME, PRENUME, TELEFON, EMAIL, ID\_ORNAMENT, ID\_CLIENTI) VALUES (13, 'Nistor', 'Adrian', '0844444555', 'adrian.nistor@example.com', 13, 13);

INSERT INTO ANGAJATI\_FLORARIE (ID\_ANGAJAT, NUME, PRENUME, TELEFON, EMAIL, ID\_ORNAMENT, ID\_CLIENTI) VALUES (14, 'Munteanu', 'Catalin', '0855555666', 'catalin.munteanu@example.com', 14, 14);

INSERT INTO ANGAJATI\_FLORARIE (ID\_ANGAJAT, NUME, PRENUME, TELEFON, EMAIL, ID\_ORNAMENT, ID\_CLIENTI) VALUES (15, 'Popa', 'Anda', '0866666777', 'anda.popa@example.com', 15, 15);

Select \* From Angajati\_Florarie;



The screenshot shows the APEX SQL Workshop interface. The SQL command window contains a series of INSERT statements for the ANGAJATI\_FLORARIE table. The results window displays a table with 15 rows, each representing an employee record.

ID_ANGAJAT	NUME	PRENUME	TELEFON	EMAIL	ID_ORNAMENT	ID_CLIENTI
4	Georgescu	Elena	734567890	elena.georgescu@example.com	3	3
6	Dumitru	Ionut	766667777	ionut.dumitru@example.com	5	5
10	Cristea	Andreea	800000111	andreea.cristea@example.com	9	9
14	Munteanu	Catalin	855555666	catalin.munteanu@example.com	14	14
12	Stoica	Bianca	833333444	bianca.stoica@example.com	12	12
13	Nistor	Adrian	844444555	adrian.nistor@example.com	13	13
2	Popescu	Ana	723456789	ana.popescu@example.com	1	1
16	Popa	Andreea	73122334	andreea.popa@example.com	5	5
5	Ionescu	Mihai	723456789	mihai.ionescu@example.com	2	2
8	Radu	Andrei	755555666	andrei.radu@example.com	4	4

## Actualizarea inregistrarilor

### 1.Sa se schimbe numele angajatului cu id=3

UPDATE ANGAJATI\_FLORARIE

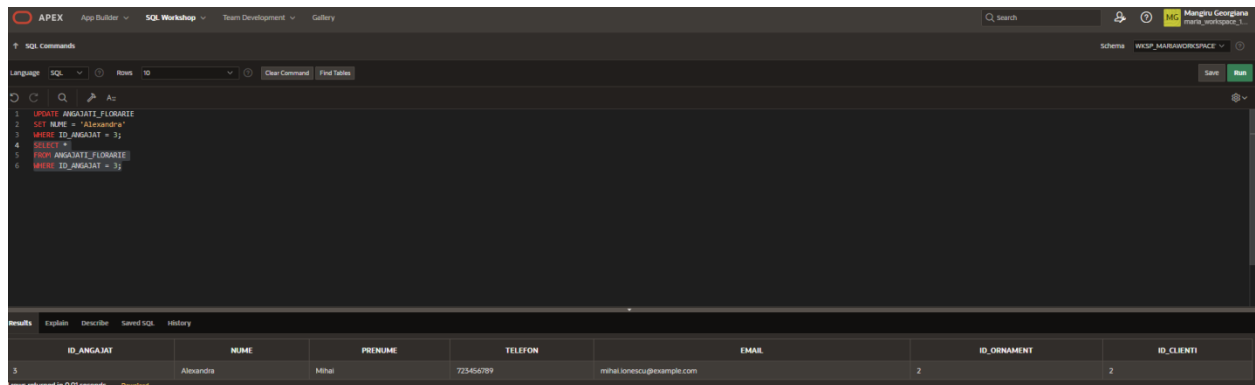
SET NUME = 'Alexandra'

WHERE ID\_ANGAJAT = 3;

SELECT \*

FROM ANGAJATI\_FLORARIE

WHERE ID\_ANGAJAT = 3;



The screenshot shows the APEX SQL Workshop interface. The SQL command window contains an UPDATE statement to change the name of the employee with ID 3 to 'Alexandra'. The results window displays a table with 1 row, representing the updated employee record.

ID_ANGAJAT	NUME	PRENUME	TELEFON	EMAIL	ID_ORNAMENT	ID_CLIENTI
3	Alexandra	Mihai	723456789	mihai.ionescu@example.com	2	2

2.Sa se actualizeze emailul clientului cu id=5.

```
UPDATE CLIENTI_FLORARIE
```

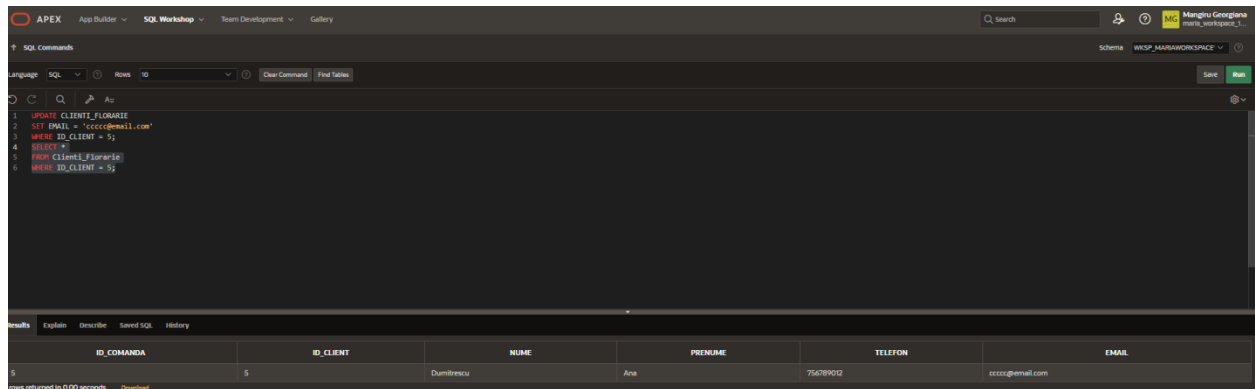
```
SET EMAIL = 'cccc@email.com'
```

```
WHERE ID_CLIENT = 5;
```

```
SELECT *
```

```
FROM Clienti_Florarie
```

```
WHERE ID_CLIENT = 5;
```



```
1 UPDATE CLIENTI_FLORARIE
2 SET EMAIL = 'cccc@email.com'
3 WHERE ID_CLIENT = 5;
4 SELECT *
5 FROM Clienti_Florarie
6 WHERE ID_CLIENT = 5;
```

ID_COMANDA	ID_CLIENT	NUME	PRENUME	TELEFON	EMAIL
5	5	Dumitrescu	Ana	756789012	cccc@email.com

1 rows returned in 0.00 seconds

3. Sa se actualizeze durata Stocurilor cu id=12

```
UPDATE STOCURI
```

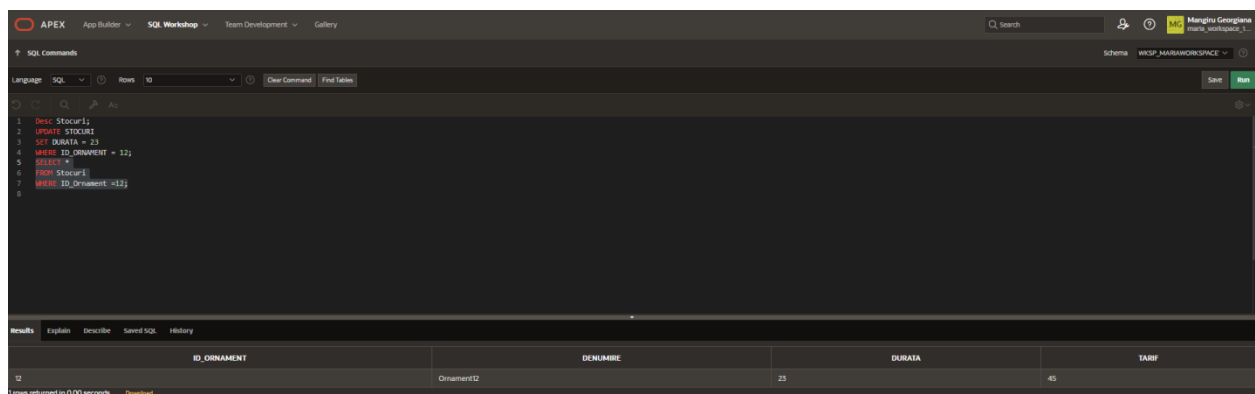
```
SET DURATA = 23
```

```
WHERE ID_ORNAMENT = 12;
```

```
SELECT *
```

```
FROM Stocuri
```

```
WHERE ID_Ornament =12;
```



```
1 Desc Stocuri;
2 UPDATE STOCURI
3 SET DURATA = 23
4 WHERE ID_ORNAMENT = 12;
5 SELECT *
6 FROM Stocuri
7 WHERE ID_Ornament =12;
```

ID_ORNAMENT	DENUMIRE	DURATA	TARIF
12	Ornament12	23	45

1 rows returned in 0.00 seconds

4.Sa se actualizeze cantitatea comenzi care are specia Crini.

```

SELECT *

FROM Comenzi_florarie

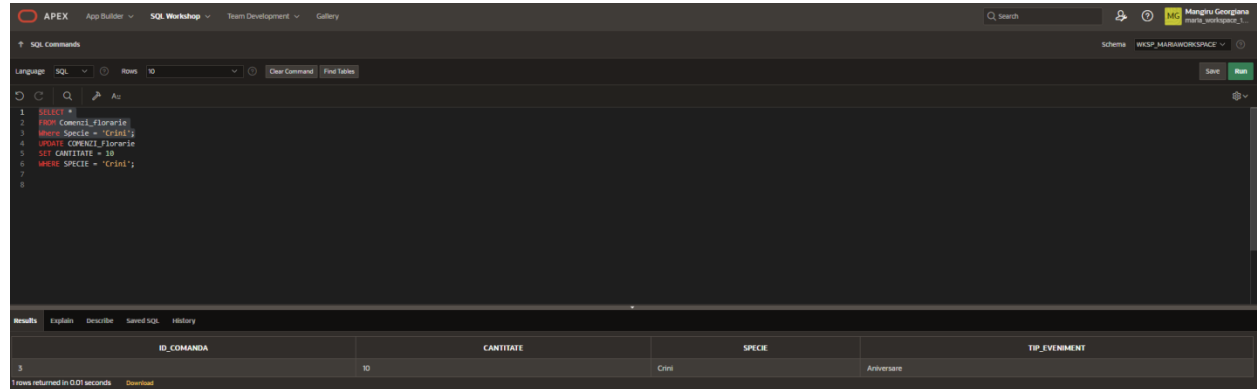
Where Specie = 'Crini';

UPDATE COMENZI_Florarie

SET CANTITATE = 10

WHERE SPECIE = 'Crini';

```



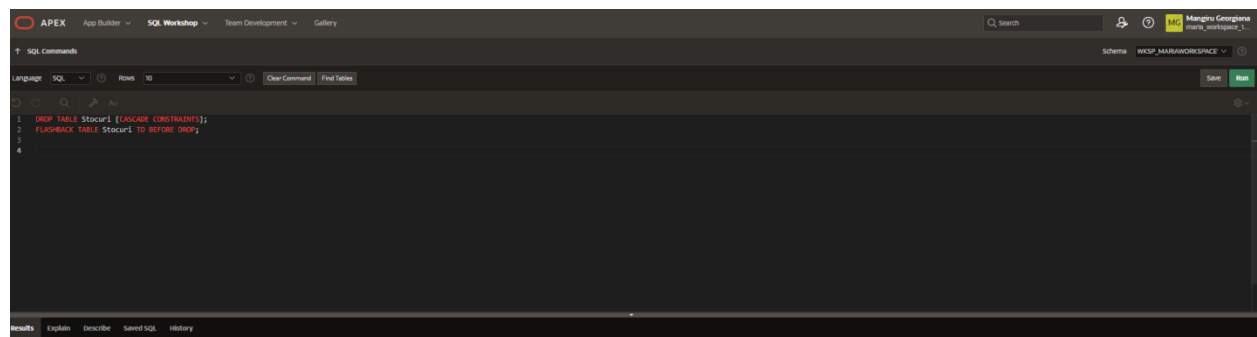
## Stergerea si recuperarea unei tabele

```

DROP TABLE Stocuri [CASCADE CONSTRAINTS];

FLASHBACK TABLE Stocuri TO BEFORE DROP;

```



Exemple de interogări variate (min 20) – incluzând și operatorii UNION, INTERSECT, MINUS, expresiile DECODE și CASE, cereri imbricate, diverse funcții single-row, functii de grup, structuri ierarhice, jonctiuni.

1.Selectează nume si prenume din tabela Angajati\_Florarie

```
SELECT Nume, Prenume FROM Angajati_Florarie;
```



The screenshot shows the APEX SQL Workshop interface. The SQL command window contains the query: `SELECT Nume, Prenume FROM Angajati_Florarie;`. The results window displays a table with two columns: **NUME** and **PRENUME**. The data is as follows:

NUME	PRENUME
Georgescu	Elena
Dumitru	Ionel
Cristea	Andreea
Murarescu	Catalin
Stoica	Bianca
Neascu	Adrian
Popescu	Ana
Popa	Andreea
Alexandru	Mihai

2.Selecteaza numele angajatiilor sau clientilor.

`SELECT Nume FROM Angajati_Florarie`

`UNION`

`SELECT Nume FROM Clienti_Florarie;`

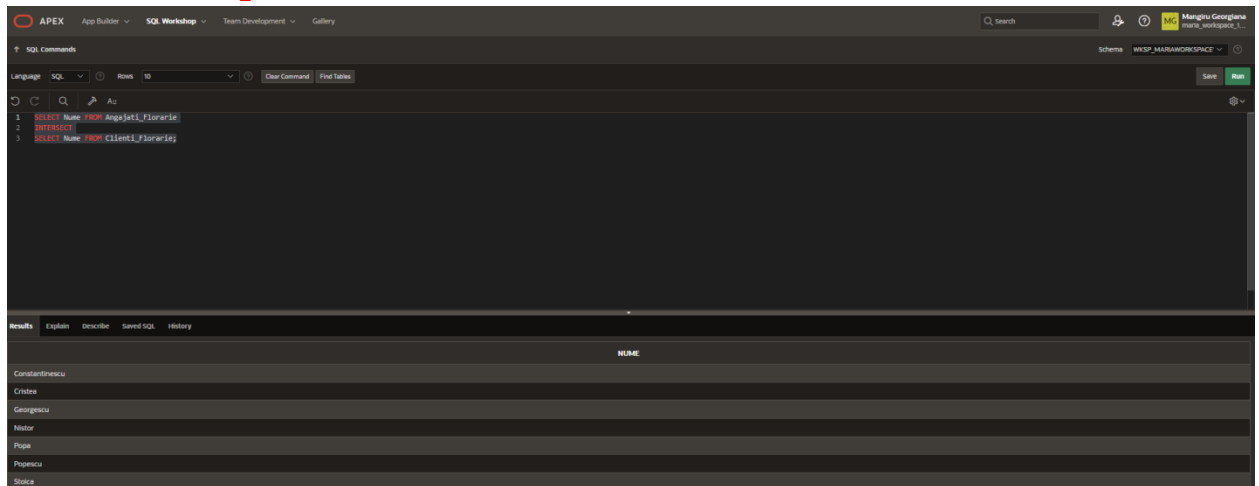
The screenshot shows the APEX SQL Workshop interface. The SQL command window contains the query: `SELECT Nume FROM Angajati_Florarie; UNION SELECT Nume FROM Clienti_Florarie;`. The results window displays a table with one column: **NUME**. The data is as follows:

NUME
Alexandru
Alexandrescu
Constantinescu
Cristea
Dumitrescu
Dumitru
Georgescu
Gheorghiu
Iancu

3.Selecteaza numele angajatiilor si a clientilor

`SELECT Nume FROM Angajati_Florarie`  
`INTERSECT`

SELECT Nume FROM Clienti\_Florarie;



The screenshot shows the APEX SQL Workshop interface. The SQL command area contains the following query:

```
1 SELECT Nume FROM Angajati_Florarie
2 INTERSECT
3 SELECT Nume FROM Clienti_Florarie;
```

The results pane shows a table with the following data:

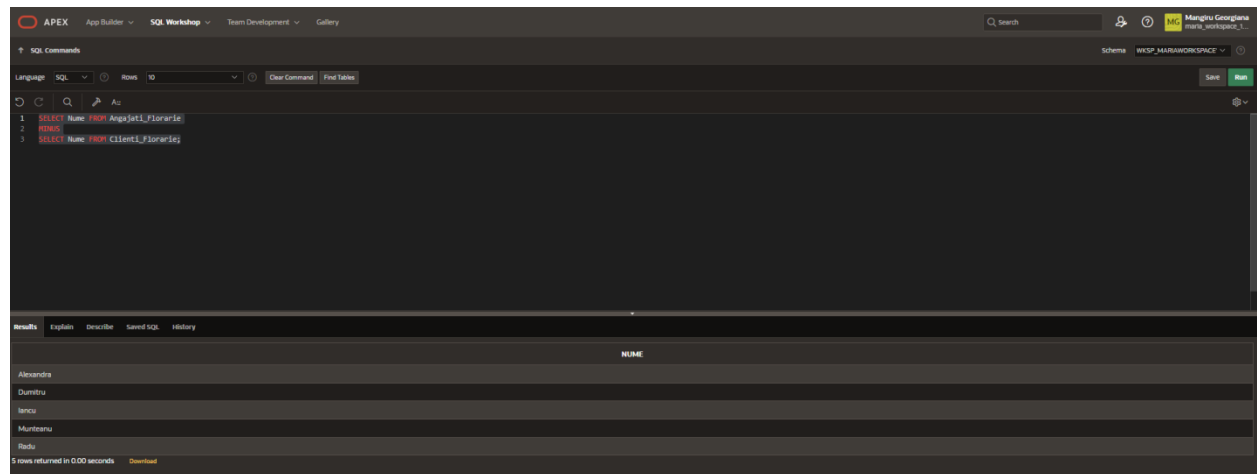
NUME
Constantinascu
Citibne
Corneanu
Holbor
Popa
Prodan
Stoica

5. Selecteaza numele angajati diferit de numele clientilor

SELECT Nume FROM Angajati\_Florarie

MINUS

SELECT Nume FROM Clienti\_Florarie;



The screenshot shows the APEX SQL Workshop interface. The SQL command area contains the following query:

```
1 SELECT Nume FROM Angajati_Florarie
2 MINUS
3 SELECT Nume FROM Clienti_Florarie;
```

The results pane shows a table with the following data:

NUME
Alexandra
Dumitru
Iancu
Munteanu
Radu

5 rows returned in 0.00 seconds

4. Sa se realizeze o clasificare a duratei de pregatire a ornamentelor

SELECT Denumire, DECODE(Durata, 20, 'Mic', 30, 'Mediu', 'Mare') AS Clasificare\_Durata FROM Stocuri;

The screenshot shows the APEX SQL Workshop interface. The SQL command area contains the following query:

```
1 SELECT Denumire, DECODE(Durata, 20, 'Mic', 30, 'Mediu', 'Mare') AS Clasificare_Durata FROM Stocuri;
```

The Results tab is active, displaying a table with two columns: DENUMIRE and CLASIFICARE\_DURATA. The table contains 10 rows of data, all with the value 'Mare' in the CLASIFICARE\_DURATA column.

DENUMIRE	CLASIFICARE_DURATA
Ornament4	Mare
Ornament3	Mare
Ornament9	Mare
Ornament10	Mare
Ornament1	Mare
Ornament14	Mare
Ornament15	Mare
Ornament13	Mare
Ornament5	Mare
Ornament2	Mare

6.Realizati o clasificare a duratei folosind functia CASE.

```
SELECT Denumire,

CASE

WHEN Durata < 60 THEN 'Mic'

WHEN Durata>= 20 AND Durata < 30 THEN 'Mediu'

ELSE 'Mare'

END AS ClasificareSalariu

FROM Stocuri;
```

The screenshot shows the APEX SQL Workshop interface. The SQL command area contains the following query:

```
1 SELECT Denumire,
2 CASE
3 WHEN Durata < 60 THEN 'Mic'
4 WHEN Durata>= 20 AND Durata < 30 THEN 'Mediu'
5 ELSE 'Mare'
6 END AS ClasificareSalariu
7 FROM Stocuri;
```

The Results tab is active, displaying a table with two columns: DENUMIRE and CLASIFICARE\_SALARIU. The table contains 10 rows of data, all with the value 'Mic' in the CLASIFICARE\_SALARIU column.

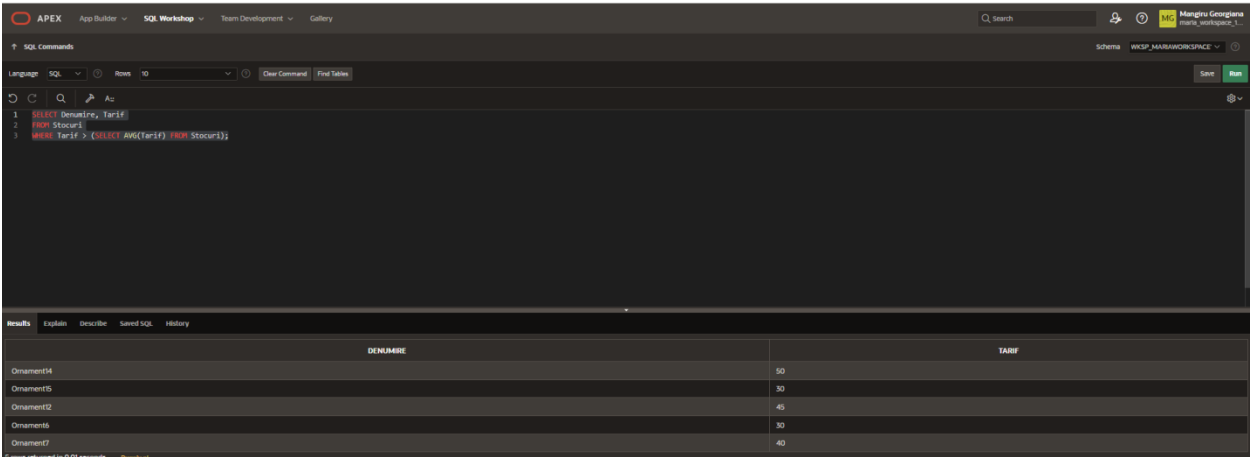
DENUMIRE	CLASIFICARE_SALARIU
Ornament4	Mic
Ornament3	Mic
Ornament9	Mic
Ornament10	Mic
Ornament1	Mic
Ornament14	Mic
Ornament15	Mic
Ornament13	Mic
Ornament5	Mic
Ornament2	Mic

7. Realizati o cerere imbracata.

```
SELECT Denumire, Tarif

FROM Stocuri
```

WHERE Tarif > (SELECT AVG(Tarif) FROM Stocuri);



The screenshot shows the APEX SQL Workshop interface. The SQL command area contains the following query:

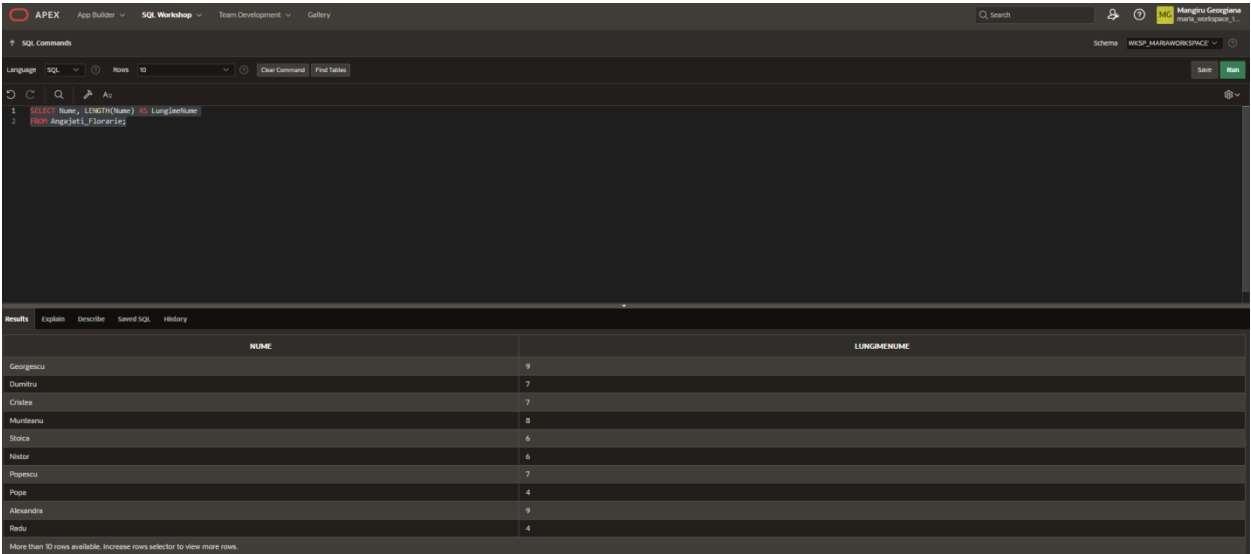
```
1 SELECT Denumire, Tarif
2 FROM Stocuri
3 WHERE Tarif > (SELECT AVG(Tarif) FROM Stocuri);
```

The results pane displays a table with two columns: DENUMIRE and TARIF. The data is as follows:

DENUMIRE	TARIF
Ornament14	50
Ornament15	30
Ornament12	45
Ornament6	30
Ornament7	40

8.Foloseste o functie single-row

SELECT Nume, LENGTH(Nume) AS LungimeNume  
FROM Angajati\_Florarie;



The screenshot shows the APEX SQL Workshop interface. The SQL command area contains the following query:

```
1 SELECT Nume, LENGTH(Nume) AS LungimeNume
2 FROM Angajati_Florarie;
```

The results pane displays a table with two columns: NUME and LUNGIMENUME. The data is as follows:

NUME	LUNGIMENUME
Georgescu	9
Dumitra	7
Cristea	7
Munteanu	8
Stanca	6
Nistor	6
Papescu	7
Popa	4
Alexandru	9
Radu	4

9. Calculati tariful mediu al ornamentului si grupati dupa denumire.

SELECT Denumire, AVG(Tarif) AS TarifMediu  
FROM Stocuri  
GROUP BY Denumire;

APEX App Builder SQL Workshop Team Development Gallery

SQL Commands

Language: SQL Rows: 10 Clear Command Find Tables

```

1 SELECT Denumire, AV(tarif) AS TarifMediu
2 FROM Stocuri
3 GROUP BY Denumire;

```

Results Explain Describe Saved SQL History

DENUMIRE	TARIFMEDIU
Ornament9	25.5
Ornament10	16
Ornament8	30
Ornament2	22.99
Ornament5	30
Ornament4	25
Ornament3	20
Ornament6	20.5
Ornament7	40
Ornament1	12.5

More than 10 rows available. Increase rows selected to view more rows.

10. Sa se afiseze numele angajatilor care au acelasi id cu cei ai clientilor.

```

SELECT
  AF.NUME AS AngajatNume,
  AF.PRENUME AS AngajatPrenume,
  AF.TELEFON AS AngajatTelefon,
  AF.EMAIL AS AngajatEmail,
  AF.ID_ORNAMENT AS AngajatIDOrnament,
  AF.SALARIU AS AngajatSalariu,
  COALESCE(CF.NUME, 'Nedeterminat') AS ClientNume,
  COALESCE(CF.PRENUME, 'Nedeterminat') AS ClientPrenume,
  COALESCE(CF.EMAIL, 'Nedeterminat') AS ClientEmail
FROM ANGAJATI_FLORARIE AF
LEFT OUTER JOIN CLIENTI_FLORARIE CF ON AF.ID_CLIENTI = CF.ID_CLIENTI;

```

APEX App Builder SQL Workshop Team Development Gallery

SQL Commands

Language: SQL Rows: 10 Clear Command Find Tables

```

1 SELECT
2   AF.NUME AS AngajatNume,
3   AF.PRENUME AS AngajatPrenume,
4   AF.TELEFON AS AngajatTelefon,
5   AF.EMAIL AS AngajatEmail,
6   AF.ID_ORNAMENT AS AngajatIDOrnament,
7   AF.SALARIU AS AngajatSalariu,
8   COALESCE(CF.NUME, 'Nedeterminat') AS ClientNume,
9   COALESCE(CF.PRENUME, 'Nedeterminat') AS ClientPrenume,
10  COALESCE(CF.EMAIL, 'Nedeterminat') AS ClientEmail
11 FROM ANGAJATI_FLORARIE AF
12 LEFT OUTER JOIN CLIENTI_FLORARIE CF ON AF.ID_CLIENTI = CF.ID_CLIENTI;

```

Results Explain Describe Saved SQL History

ANGAJATNUME	ANGAJATPRENUME	ANGAJATELEFON	ANGAJATEMAIL	ANGAJATIDORNAMENT	ANGAJATSALARIU	CLIENTNUME	CLIENTPRENUME	CLIENTEMAIL
Georgescu	Elena	754567890	elena.georgescu@example.com	3	-	Georgescu	Elena	georgescu@example.com
Dumitru	Ionut	76666777	ionut.dumitru@example.com	5	-	Dumitrescu	Ana	cccc@gmail.com
Popa	Andreea	7312334	andreea.popa@example.com	3	-	Dumitrescu	Ana	cccc@gmail.com
Alexandra	Mihai	723456789	mihai.ionescu@example.com	2	-	Ionescu	Mihai	ionescu@example.com
Stoici	Bianca	83333444	bianca.stoici@example.com	10	-	Constantinescu	Raluca	constantin@example.com
Popescu	Ana	712345678	ana.popescu@example.com	1	-	Popescu	Ana	popescu@example.com
Radu	Andrei	75555666	andrei.radu@example.com	4	-	Vasilescu	Ion	vasilescu@example.com
Iancu	Cristina	788889999	cristina.iancu@example.com	7	-	Gheorghiu	Adrian	gheorghiu@example.com
Cristea	Andreea	80000111	andreea.cristea@example.com	9	-	Alexandrescu	George	alexandr@example.com
Restor	Adrian	84444555	adrian.restor@example.com	15	-	Iordache	Silviu	iordache@example.com

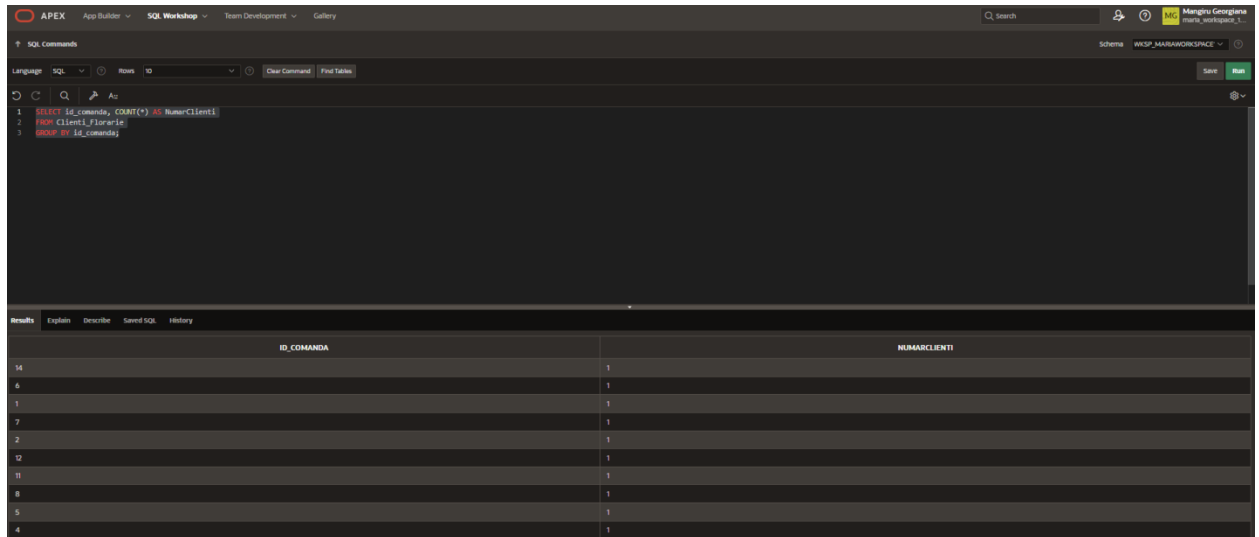
11..Aflati cate comenzi are un client.

```

SELECT id_comanda, COUNT(*) AS NumarClienti
FROM Clienti_Florarie

```

GROUP BY id\_comanda;



The screenshot shows the APEX SQL Workshop interface. The SQL command area contains the following query:

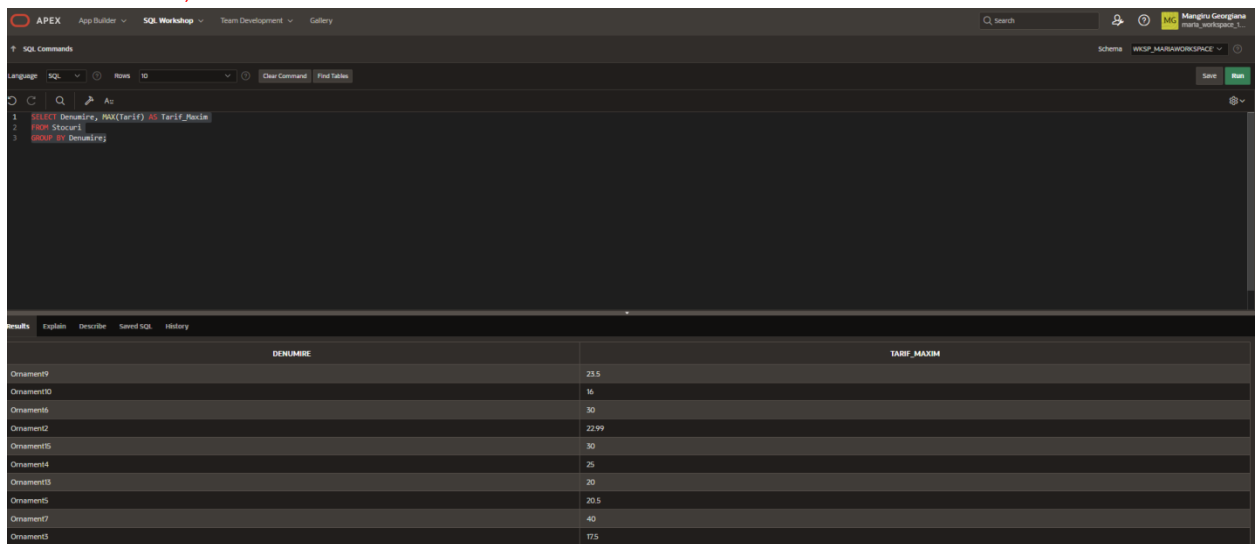
```
1 SELECT id_comanda, COUNT(*) AS NumarClienti
2 FROM Clienti_fiorarie
3 GROUP BY id_comanda;
```

The results table has two columns: ID\_COMANDA and NUMARCLIENTI. The data is as follows:

ID_COMANDA	NUMARCLIENTI
14	1
6	1
1	1
7	1
2	1
12	1
10	1
8	1
5	1
4	1

12. Aflati tariful maxim grupand dupa denumire.

SELECT Denumire, MAX(Tarif) AS Tarif\_Maxim  
FROM Stocuri  
GROUP BY Denumire;



The screenshot shows the APEX SQL Workshop interface. The SQL command area contains the following query:

```
1 SELECT Denumire, MAX(Tarif) AS Tarif_Maxim
2 FROM Stocuri
3 GROUP BY Denumire;
```

The results table has two columns: DENUMIRE and TARIF\_MAXIM. The data is as follows:

DENUMIRE	TARIF_MAXIM
Ornament9	23.5
Ornament10	16
Ornament6	30
Ornament2	22.99
Ornament5	30
Ornament4	25
Ornament3	20
Ornament5	20.5
Ornament7	40
Ornament3	17.5

13. Aflati durata totala de a crea un ornament si grupati dupa denumire.

SELECT Denumire, SUM(Durata) AS Total\_Durata  
FROM Stocuri  
GROUP BY Denumire;

APEX App Builder SQL Workshop Team Development Gallery

SQL Commands

Language: SQL Rows: 10 Clear Command Find Tables Save Run

```

1 SELECT Denumire, SUM(Durata) AS Total_Durata
2 FROM Stocuri
3 GROUP BY Denumire;

```

Results Explain Describe Saved SQL History

DENUMIRE	TOTAL_DURATA
Ornament9	32
Ornament10	19
Ornament8	45
Ornament2	40
Ornament5	40
Ornament4	35
Ornament3	29
Ornament6	28
Ornament7	50
Ornament1	22

14. Sa se afiseze ornamentele care depasesc un tariff de 20.

```

WITH Ornamente_cu_Tarif_mare AS (
    SELECT Denumire, Tarif
    FROM Stocuri
    WHERE Tarif > 20
)
SELECT * FROM Ornamente_cu_Tarif_mare;
Select * FROM Stocuri;

```

APEX App Builder SQL Workshop Team Development Gallery

SQL Commands

Language: SQL Rows: 10 Clear Command Find Tables Save Run

```

1 WITH Ornamente_cu_Tarif_mare AS (
2     SELECT Denumire, Tarif
3     FROM Stocuri
4     WHERE Tarif > 20
5 )
6 SELECT * FROM Ornamente_cu_Tarif_mare;
Select * FROM Stocuri;

```

Results Explain Describe Saved SQL History

DENUMIRE	TARIF
Ornament4	25
Ornament9	23.5
Ornament14	50
Ornament5	30
Ornament5	20.5
Ornament12	45
Ornament6	30
Ornament2	22.99
Ornament7	40

15.Functia Rownum pentru tabela Comenzi

```

SELECT Specie, Cantitate
FROM Comenzi_Florarie
WHERE ROWNUM <= 5;

```

The screenshot shows the APEX SQL Workshop interface. The SQL command window contains the following query:

```

1 SELECT Specie, Cantitate
2 FROM Comenzi_Florarie
3 WHERE REGEXP_LIKE(Specie, '^C');

```

The results window displays the following data:

SPECIE	CANTITATE
Trandafiri	10
Geraniu rosi	5
Chimonilla	3
Trandafiri galbeni	5
Cirese	10

16. Sa se afiseze din tabela Comenzi speciile de flori care incep cu litera “c”

SELECT Specie

FROM Comenzi\_Florarie

WHERE REGEXP\_LIKE(Specie, '^C');

The screenshot shows the APEX SQL Workshop interface. The SQL command window contains the following query:

```

1 SELECT Specie
2 FROM Comenzi_Florarie
3 WHERE REGEXP_LIKE(Specie, '^C');

```

The results window displays the following data:

SPECIE
Cirese
Cirese roz
Cirese albe

3 rows returned in 0.01 seconds

17. Utilizati functia NVL pentru gestionarea valorilor nule.

SELECT Specie, NVL(Cantitate, 0) AS Specie\_inexistenta

FROM Comenzi\_Florarie;



APEX App Builder SQL Workshop Team Development Gallery

SQL Commands

```

1 SELECT Specie, NVL(Cantitate, 0) AS Specie_inexistenta
2 FROM Comenzi_Florarie;

```

Results Explain Describe Saved SQL History

SPECIE	SPECIE_INEXISTENTA
Trandafiri	10
Geraniole rosi	6
Orhidee roz	5
Trandafiri galbene	5
Crimi	10
Laletile roz	5
Geraniole galbene	4
Orhidee	8
Bugeti roz	4
Trandafiri albi	2

18. Utilizarea funcției analytic ROW\_NUMBER() pentru tabela Stocuri.

```

SELECT Specie,Cantitate ,
       ROW_NUMBER() OVER (PARTITION BY Specie ORDER BY Cantitate DESC) AS Rang_In_Comenzi
FROM Comenzi_Florarie;

```

APEX App Builder SQL Workshop Team Development Gallery

SQL Commands

```

1 SELECT Specie,Cantitate ,
2       ROW_NUMBER() OVER (PARTITION BY Specie ORDER BY Cantitate DESC) AS Rang_In_Comenzi
3 FROM Comenzi_Florarie;

```

Results Explain Describe Saved SQL History

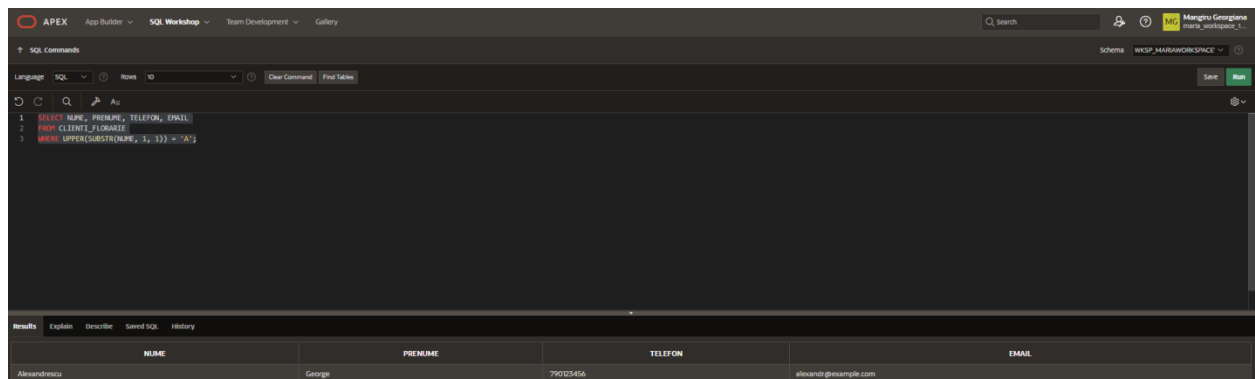
SPECIE	CANTITATE	RANG_IN_COMENZI
Bugeti albi	6	1
Bugeti roz	4	1
Crimi	10	1
Crimi albi	8	1
Crimi roz	7	1
Geraniole galbene	4	1
Geraniole rosi	6	1
Laletile	15	1
Laletile roz	5	1
Orhidee	8	1

19. Obține informații despre clienții cu nume care încep cu litera 'A'

```

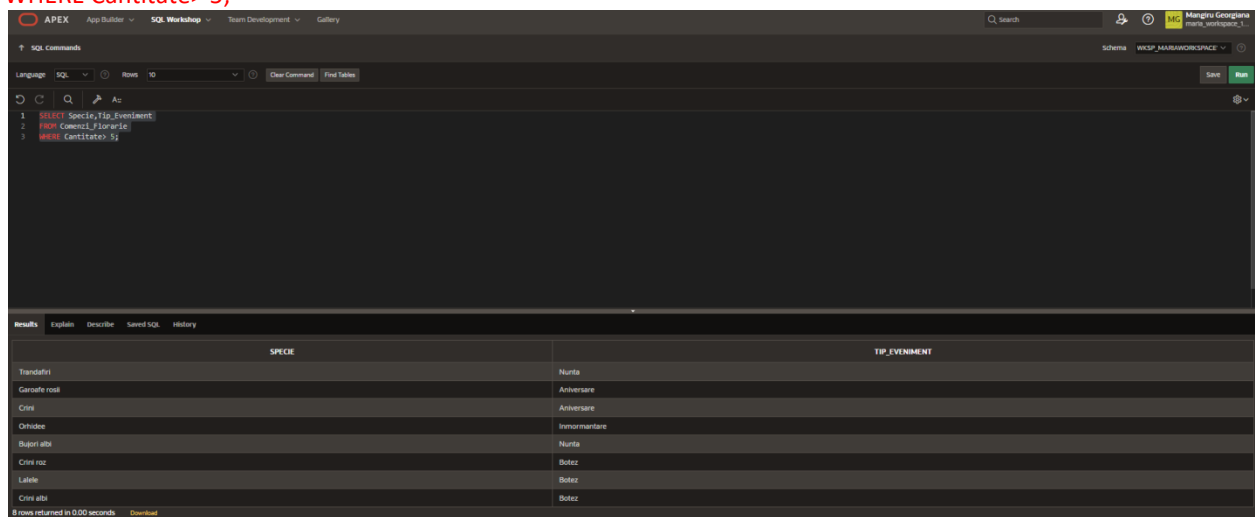
SELECT NUME, PRENUME, TELEFON, EMAIL
FROM CLIENTI_FLORARIE
WHERE UPPER(SUBSTR(NUME, 1, 1)) = 'A';

```



20. Obțineți informații despre comenzile a carui cantitate depășește 5 buc.

```
SELECT Specie, Tip_Eveniment
FROM Comenzi_Florarie
WHERE Cantitate > 5;
```



21. Calculați discountul (DC) pentru clienți în funcție de următoarele condiții:

Dacă un client a încheiat o comandă, atunci discountul (DC) va fi de 10%;

Dacă un client a încheiat două comenzi, atunci discountul (DC) va fi de 15%;

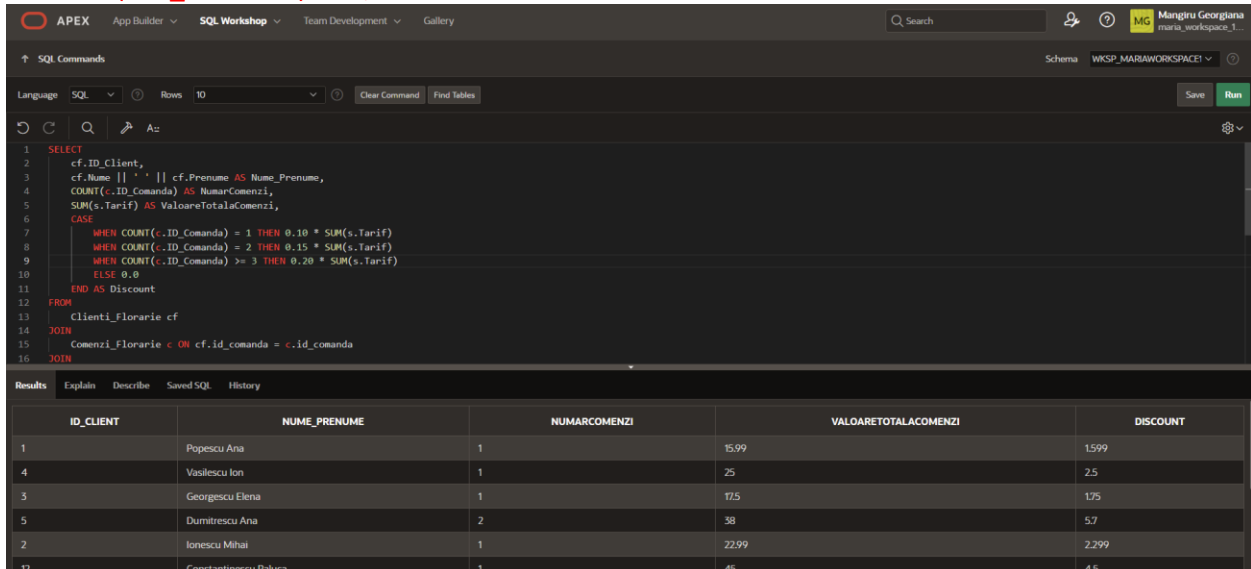
Dacă un client a încheiat mai mult de trei comenzi, atunci discountul (DC) va fi de 20%.

```
SELECT
  cf.ID_Client,
  cf.Nume || ' ' || cf.Prenume AS Nume_Prenume,
  COUNT(c.ID_Comanda) AS NumarComenzi,
  SUM(s.Tarif) AS ValoareTotalaComenzi,
  CASE
    WHEN COUNT(c.ID_Comanda) = 1 THEN 0.10 * SUM(s.Tarif)
    WHEN COUNT(c.ID_Comanda) = 2 THEN 0.15 * SUM(s.Tarif)
    WHEN COUNT(c.ID_Comanda) >= 3 THEN 0.20 * SUM(s.Tarif)
    ELSE 0.0
  END AS Discount
FROM
```

```

    Clienti_Florarie cf
JOIN
    Comenzi_Florarie c ON cf.id_comanda = c.id_comanda
JOIN
    Angajati_Florarie a ON a.ID_Clienti = cf.ID_Client
JOIN Stocuri s ON s.id_ornament=a.id_ornament
GROUP BY
    cf.ID_Client, cf.Nume, cf.Prenume
HAVING
    COUNT(c.ID_Comanda) >= 1;

```



The screenshot shows the APEX SQL Workshop interface. The SQL command window contains a query that selects client information, command counts, and discounts based on the number of commands. The results window displays a table with 5 columns: ID\_CLIENT, NUME\_PRENUME, NUMARCOMENZI, VALOARETOTALACOMENZI, and DISCOUNT. The results are sorted by ID\_CLIENT.

ID_CLIENT	NUME_PRENUME	NUMARCOMENZI	VALOARETOTALACOMENZI	DISCOUNT
1	Popescu Ana	1	15.99	1.599
4	Vasilescu Ion	1	25	2.5
3	Georgescu Elena	1	17.5	1.75
5	Dumitrescu Ana	2	38	5.7
2	Ionescu Mihai	1	22.99	2.299
17	Constantinescu Dăbura	1	45	4.5

22. Se calculeze diferit sporul de munca pentru angajati astfel:

- daca au de facut ornamente de nunta li se acorda un spor de 30%;
- daca au de facut ornamente de botez 20%
- daca au de facut ornamente de aniversare 15% Din acestea sa se elimine inregistrarile incheiate de clientii care incep cu litera M.

Ordonati descrescator in functie de numele angajatilor

```

SELECT
    af.Nume AS Nume_Angajat,
    af.Prenume AS Prenume_Angajat,
CASE
    WHEN cl.Tip_Eveniment = 'Nunta' THEN s.Tarif * 0.3
    WHEN cl.Tip_Eveniment = 'Botez' THEN s.Tarif * 0.2
    WHEN cl.Tip_Eveniment = 'Aniversare' THEN s.Tarif * 0.15
    ELSE 0
END AS Spor_Munca
FROM
    Angajati_Florarie af
JOIN
    Clienti_Florarie cf ON af.ID_Clienti = cf.ID_Client
JOIN
    Comenzi_Florarie cl ON cf.ID_Comanda = cl.ID_Comanda

```

JOIN

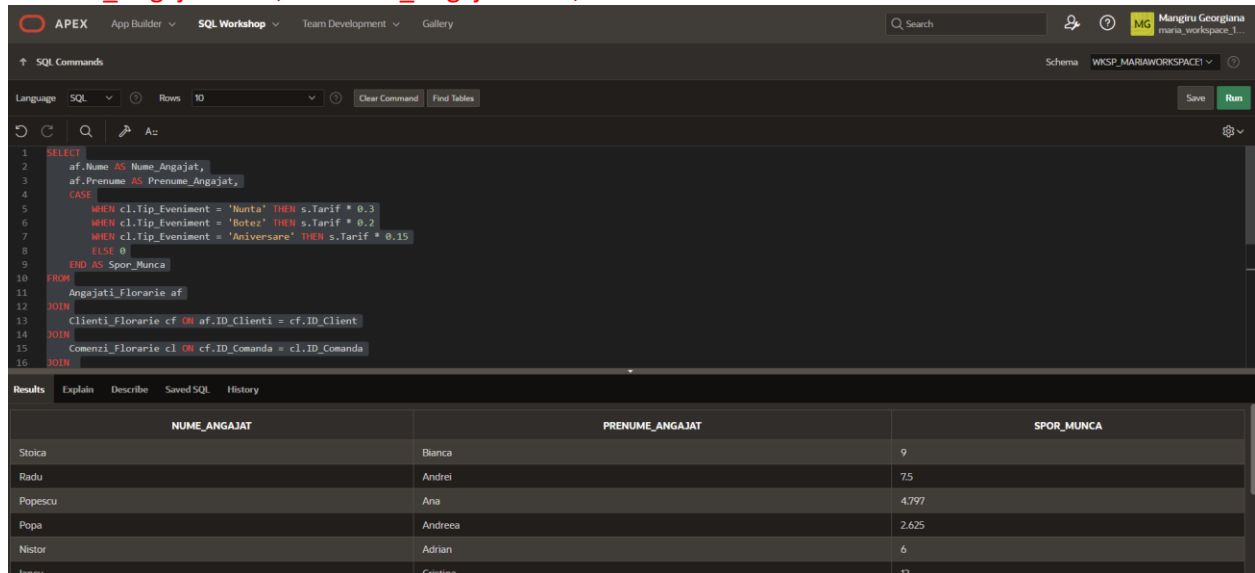
Stocuri s ON s.ID\_Ornament = af.ID\_Ornament

WHERE

NOT af.Nume LIKE 'M%'

ORDER BY

Nume\_Angajat DESC, Prenume\_Angajat DESC;



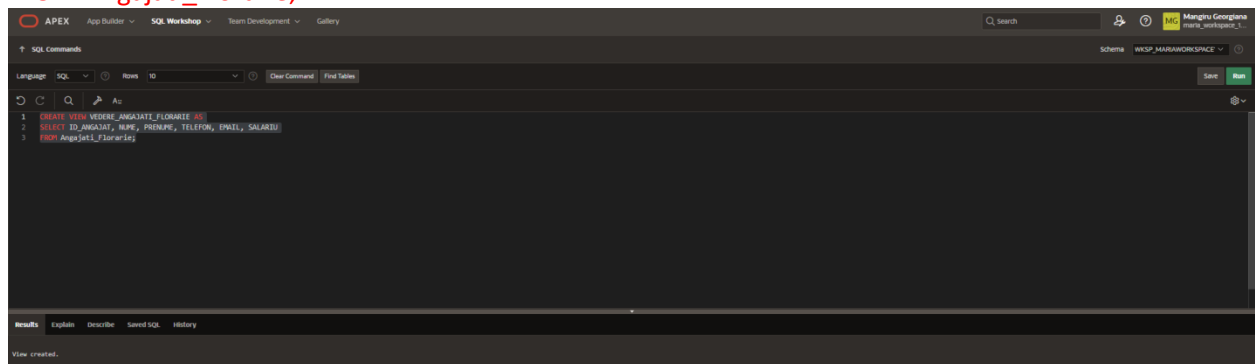
The screenshot shows the APEX SQL Workshop interface. The SQL command window contains a query that selects employee names and calculates a bonus based on their event type. The results window shows a table with three columns: NUME\_ANGAJAT, PRENUME\_ANGAJAT, and SPOR\_MUNCA. The data is as follows:

NUME_ANGAJAT	PRENUME_ANGAJAT	SPOR_MUNCA
Stoica	Bianca	9
Radu	Andrei	75
Popescu	Ana	4797
Popa	Andreea	2.625
Nistor	Adrian	6
Ioncu	Cristina	12

Gestiunea altor obiecte ale bazei de date: vederi, indecsi, sinonime, secvente

Crearea unei vederi care să afișeze informații din tabela Angajati\_Florarie:

```
CREATE VIEW VEDERE_ANGAJATI_FLORARIE AS
SELECT ID_ANGAJAT, NUME, PRENUME, TELEFON, EMAIL, SALARIU
FROM Angajati_Florarie;
```



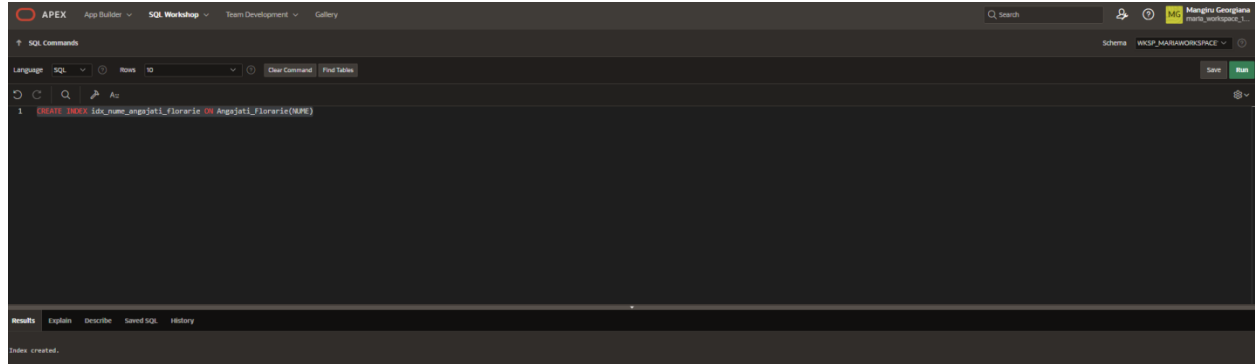
The screenshot shows the APEX SQL Workshop interface with the SQL command window containing the following code:

```
1 CREATE VIEW VEDERE_ANGAJATI_FLORARIE AS
2 SELECT ID_ANGAJAT, NUME, PRENUME, TELEFON, EMAIL, SALARIU
3 FROM Angajati_Florarie;
```

The results window shows the message: View created.

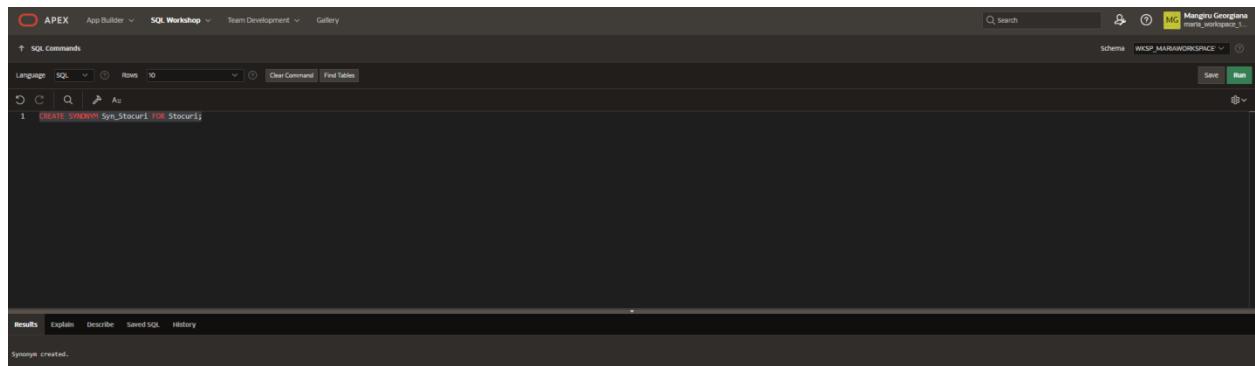
Crearea unui index pe coloana NUME din tabela Angajati\_Florarie

CREATE INDEX idx\_ume\_angajati\_florarie ON Angajati\_Florarie(NUME)



Crearea unui sinonim pentru a oferi o denumire alternativă pentru tabela Stocuri.

CREATE SYNONYM Syn\_Stocuri FOR Stocuri;



Crearea unei secvențe pentru a genera valori unice.

CREATE SEQUENCE Seq\_ID\_Client START WITH 1 INCREMENT BY 1;

