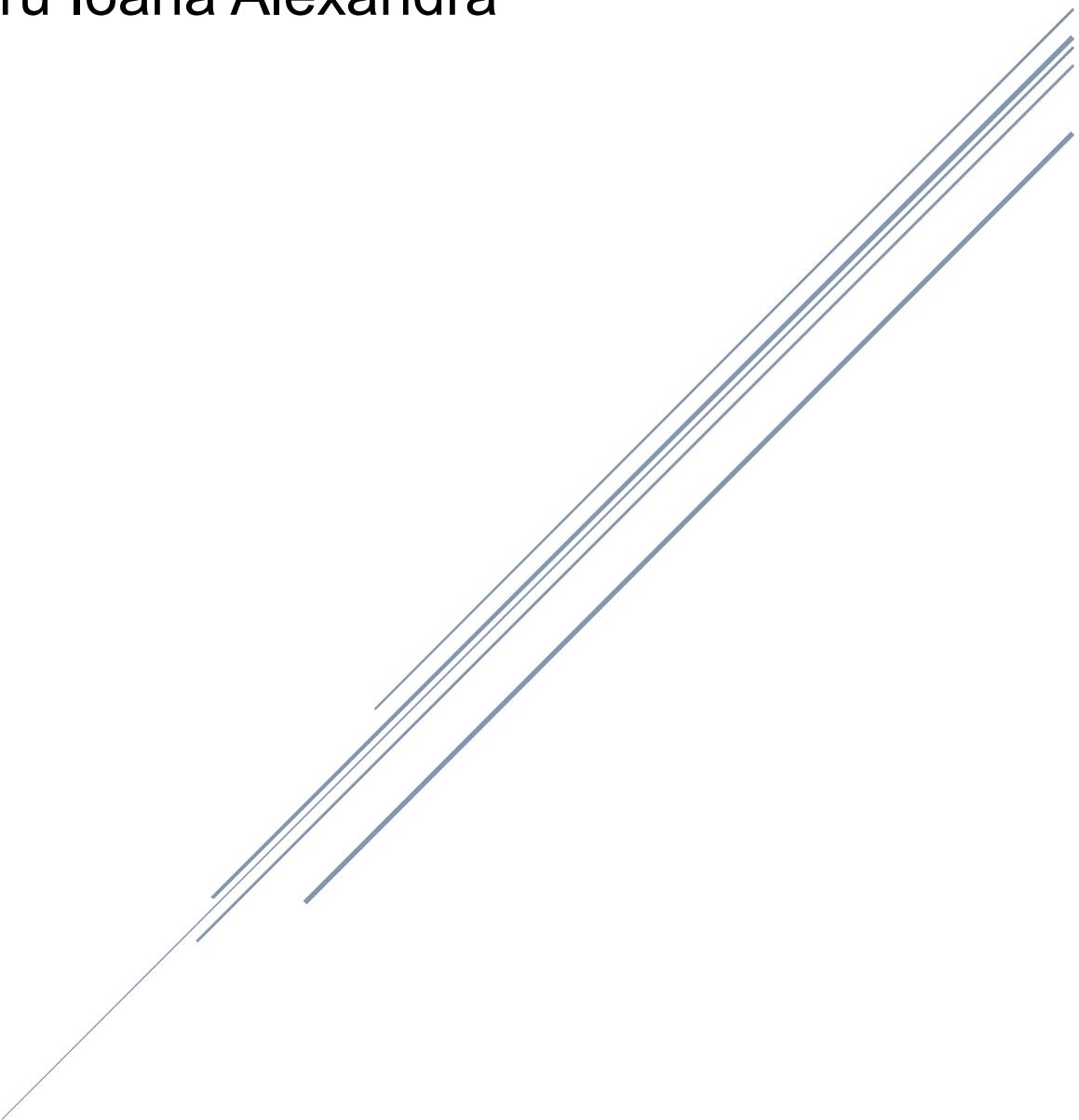


Sistemul de gestiune al cinematografelor din România

Tunaru Ioana Alexandra



Seria 23, Grupa 232
2024-2025

Cuprins

Introducere	3
1. Prezentarea bazei de date	4
2. Diagrama ERD	5
3. Diagrama Conceptuală	6
4. Definirea tabelelor	7
5. Inserare date în tabele	11
ORAS	11
ADRESA	12
CINEMATOGRAF	13
DEPARTAMENTE	14
JOBURI	15
ANGAJATI	16
FILM	18
CLIENTI	19
RECENZIE	20
REZERVARE	22
DIFUZEAZA	23
6. Procedură care utilizează cele 3 tipuri de colecții studiate	30
Testare procedură ex6	34
7. Procedură care utilizează cursor dinamic	35
Testare procedura ex7	37
8. Funcție care utilizează tabel auxiliar pentru gestionarea erorilor	40
Testare functie ex8	44
9. Procedură care utilizează 4 parametrii și 3 exceptii proprii	50
Testare procedura ex9	54
10. Trigger de tip LMD la nivel de comandă	61
Testare trigger_ex10	65
-- verificare inserări	65
-- verificare update	67
-- verificare delete	68

11.	Trigger de tip LMD la nivel de linie (compound trigger).....	70
	Testare trigger_ex11	73
12.	Trigger de tip LDD	80
	Testare trigger_ex12	81
13.	Pachet complex (se utilizează DBMS_JOB)	86
	Testare pachet_ex13	97
	Pasul 1.....	97
	Pasul 2.....	102
	Pasul 3.....	103
	Pasul 4.....	105
	Pasul 5.....	107
	Pasul 6.....	108

Introducere

Cinematografele joacă un rol important în viața culturală și de divertisment a comunităților. Pentru a gestiona eficient activitatea acestora, este necesar un sistem bine organizat, care să permită administrarea informațiilor despre locații, angajați, clienți și filme.

Astfel, baza de date propusă va centraliza și structura informații esențiale, incluzând detalii despre locații, programe, rezervări și recenzii. De asemenea, va oferi facilități pentru gestionarea salariilor angajaților, fidelizarea clientilor și monitorizarea performanței filmelor. Acest sistem își propune să asigure o administrare eficientă, transparentă și orientată spre satisfacerea nevoilor publicului.

Implementarea proiectului s-a realizat utilizând Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 ca sistem de gestiune a bazelor de date. Aplicația SQL Developer a fost rulată pe un sistem fizic echipat cu 16 GB RAM, dintre care aproximativ 1.16 GB au fost alocați în timpul execuției. Sistemul de operare utilizat este Windows 11 Pro, iar proiectul a fost desfășurat direct pe hardware-ul local, fără a recurge la o mașină virtuală.

1. Prezentarea bazei de date

Baza de date va fi proiectată pentru a gestiona informații despre cinematografele din diverse orașe din România. Fiecare **cinematograf** este asociat unei **adrese unice**, care include detalii despre **orașul în care se află**. Se vor înregistra informații despre capacitatea sălilor și compania care administrează cinematograful.

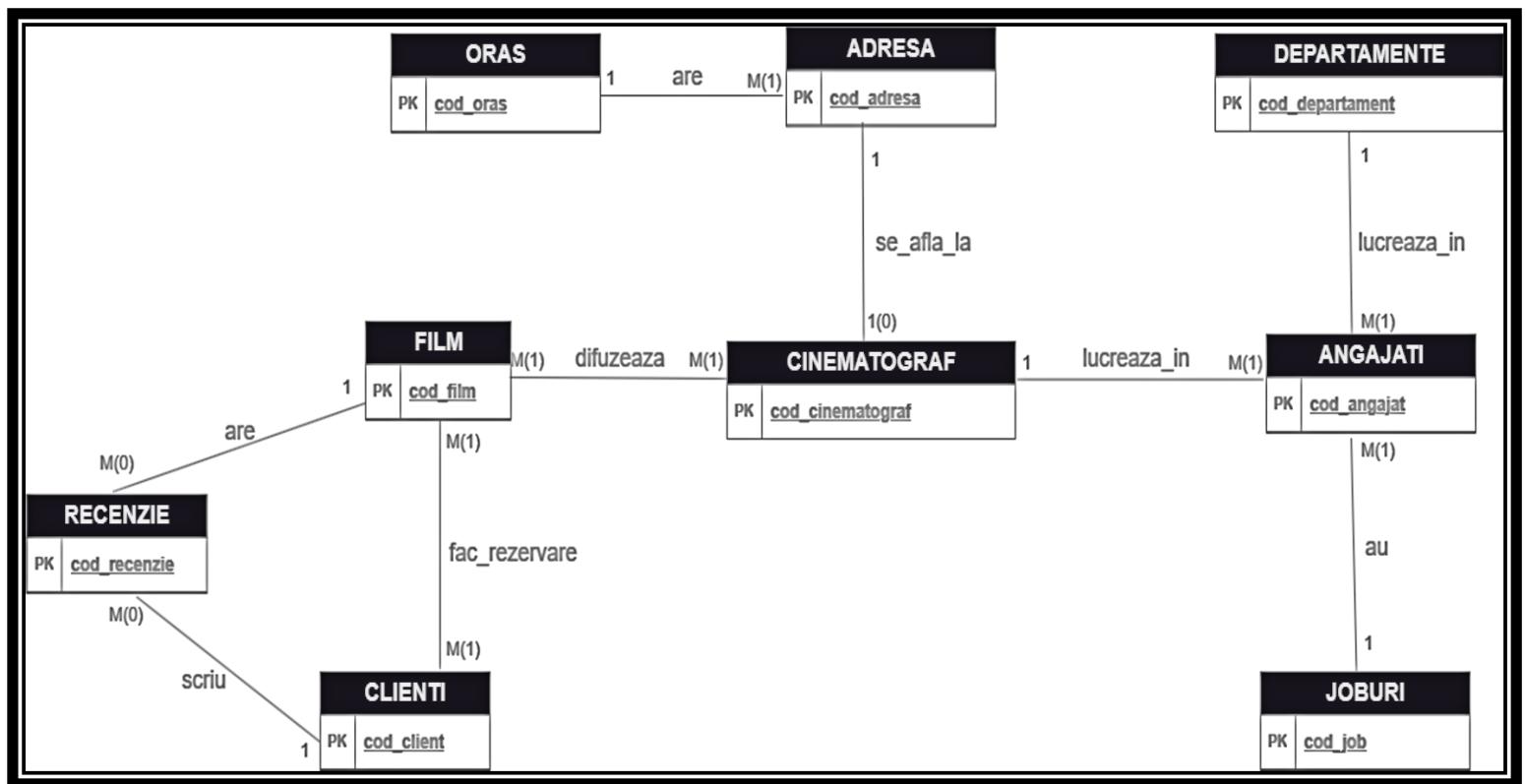
Angajații cinematografelor sunt organizați pe **departamente**, fiecare având un singur **job** și lucrând într-un singur cinematograf. Pentru fiecare angajat vor fi înregistrate informații precum statusul (activ, în concediu sau suspendat), data angajării și salariul. Salariul trebuie să se încadreze între valorile minime și maxime stabilite pentru postul respectiv, în funcție de nivelul de experiență necesar.

Clienții cinematografelor pot face mai multe **rezervări** și beneficiază de reduceri printr-un sistem de fidelizare bazat pe puncte. Fiecare rezervare în valoare de X lei aduce X puncte. Punctele acumulate se transformă în reduceri: de exemplu 200 de lei aduc o reducere de 20 de lei la următoarea rezervare. Fiecare rezervare include un singur **film** ales de client. Prețurile biletelor variază în funcție de formatul proiecției: 25 de lei pentru 2D, 35 de lei pentru 3D și 50 de lei pentru IMAX.

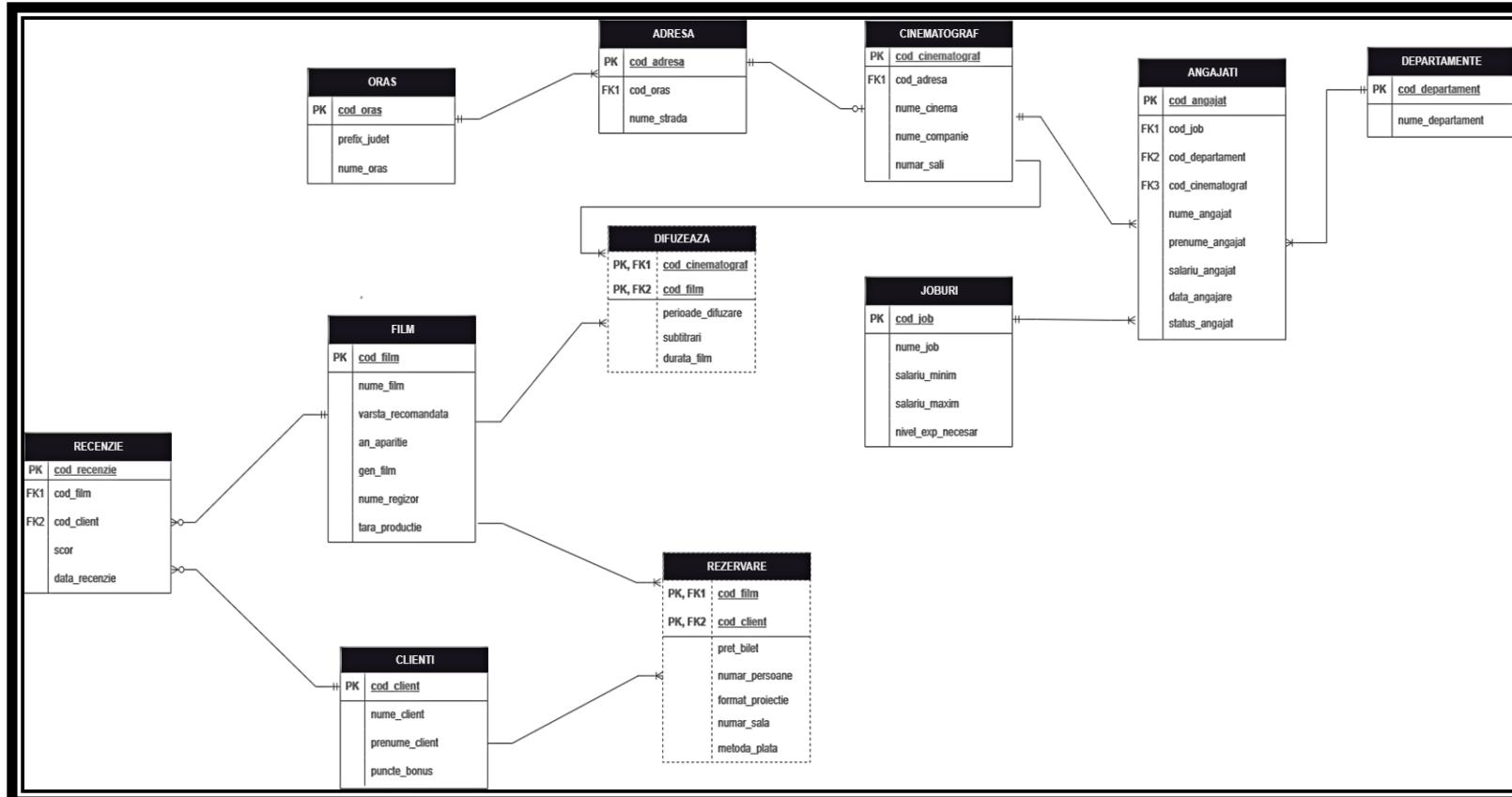
Cinematografele vor afișa un program detaliat al filmelor, fiecare având opțiuni de subtitrare (cele mai frecvente fiind în română și engleză), un număr specific de locuri disponibile în săli și o perioadă de **difuzare** organizată pe zile și intervale orare. Fiecare film va avea o restricție de vîrstă corespunzătoare.

Clienții pot oferi **recenzii** pentru filmele vizionate, contribuind la evaluări detaliate și relevante. Acest sistem este conceput pentru a oferi o experiență eficientă și personalizată, atât în administrarea cinematografelor, cât și în creșterea satisfacției clienților.

2. Diagrama ERD



3. Diagrama Conceptuală



4. Definirea tablelor

```
CREATE TABLE oras(  
cod_oras number(3) constraint pk_oras primary key,  
prefix_judet varchar2(3) not null,  
nume_oras varchar2(30) not null);
```

```
CREATE TABLE adresa (  
cod_adresa number(3) constraint pk_adresa primary key,  
cod_oras number(3),  
nume_strada varchar2(30) not null,  
constraint fk_oras foreign key(cod_oras) references ORAS(cod_oras));
```

```
CREATE TABLE cinematograf(  
cod_cinematograf varchar2(3) constraint pk_cinema primary key,  
cod_adresa number(3),  
nume_cinema varchar2(30) not null,  
nume_companie varchar2(30) not null,  
numar_sali number(3) not null,  
constraint fk_adresa foreign key(cod_adresa) references ADRESA(cod_adresa));
```

```
CREATE TABLE departamente(  
cod_departament varchar2(3) constraint pk_dep primary key,  
nume_departament varchar2(30) not null);
```

```
CREATE TABLE joburi(  
cod_job varchar2(3) constraint pk_job primary key,  
nume_job varchar2(30) not null,  
salariu_minim number(6) not null,  
salariu_maxim number(6) not null,  
nivel_exp_necesar varchar2(30));
```

```
CREATE TABLE angajati(  
cod_angajat varchar2(3) constraint pk_angajat primary key,  
cod_job varchar2(3),  
cod_departament varchar2(3),  
cod_cinematograf varchar2(3),
```

```

nume_angajat varchar2(30) not null,
prenume_angajat varchar2(30) not null,
salariu_angajat number(6) not null,
data_angajare date,
status_angajat varchar2(30),
constraint fk_job foreign key(cod_job) references JOBURI(cod_job),
constraint fk_dep foreign key(cod_departament) references
DEPARTAMENTE(cod_departament),
constraint fk_cinema foreign key (cod_cinematograf) references
CINEMATOGRAF(cod_cinematograf)
);

```

```

CREATE TABLE film(
cod_film varchar2(3) constraint pk_film primary key,
nume_film varchar2(30) not null,
varsta_recomandata number(2) not null,
an_aparitie number(4),
gen_film varchar2(30) not null,
nume_regizor varchar2(30),
tara_productie varchar2(30));

```

```

CREATE TYPE per_dif AS OBJECT (
  data_difuzare date,
  ora_inceput date,
  ora_final date
);
/

```

```

CREATE TYPE lista_perioade AS TABLE OF per_dif;
/

```

```

CREATE TABLE difuzeaza(
cod_cinematograf varchar2(3) references CINEMATOGRAF(cod_cinematograf),
cod_film varchar2(3) references FILM(cod_film),
perioade_difuzare lista_perioade,
subtitrari varchar2(30),
durata_film number(3),
constraint pk_difuzare primary key (cod_cinematograf, cod_film)
) NESTED TABLE perioade_difuzare STORE AS perioade_difuzare_tab;

```

```
CREATE TABLE clienti(
cod_client number(2) constraint pk_client primary key,
nume_client varchar2(30) not null,
prenume_client varchar2(30) not null,
puncte_bonus number(5));
```

```
CREATE TABLE recenzie(
cod_recenzie varchar(3) constraint pk_recenzie primary key,
cod_film varchar2(3),
cod_client number(2),
scor number(2),
data_recenzie date,
constraint fk_film_recenzie foreign key(cod_film) references FILM(cod_film),
constraint fk_client foreign key(cod_client) references CLIENTI(cod_client));
```

```
CREATE TABLE rezervare(
cod_film varchar2(3) references FILM(cod_film),
cod_client number(2) references CLIENTI(cod_client),
pret_bilet number(3) not null,
numar_persoane number(2) not null,
format_proiectie varchar2(5) not null,
numar_sala number(3) not null,
metoda_plata varchar2(30) not null,
constraint pk_rezervare primary key (cod_film, cod_client));
```

The screenshot shows the Oracle SQL Developer interface with the following details:

- Title Bar:** Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\CREATE+INSERTARE_TABLELE.sql
- Menu Bar:** File, Edit, View, Navigate, Run, Source, Team, Tools, Window, Help
- Toolbar:** Standard toolbar with icons for New, Open, Save, Print, etc.
- Project Explorer:** Shows Oracle Connect, BD, PROJECT, SGBD, Oracle NoSQL DB, and Database Schema.
- Worksheet:** Contains the SQL code for creating the ORAS table:

```
CREATE TABLE oras(
cod_oras number(3) constraint pk_oras primary key,
prefix_judet varchar2(3) not null,
nume_oras varchar2(30) not null);
```
- Script Output:** Displays the results of the table creation:

```
Table ORAS created.
```
- Query Result:** Displays the results of the table creation:

```
Table ADRESA created.
```
- Structure:** Shows the table structure for ORAS.

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\CREATE+INSERARE_TABLE.sql

File Edit View Navigate Run Source Team Tools Window Help

Oracle Connect PROJECT

CREATE+INSERARE_TABLE.sql Ex 6.sql Ex 7.sql Ex 8.sql Ex 9.sql Ex 10.sql Ex 11.sql Ex 12.sql Ex 13.sql

SQL Worksheet History

Worksheet Query Builder

```
CREATE TABLE angajati(
cod_angajat varchar2(3) constraint pk_angajat primary key,
cod_job varchar2(3),
cod_departament varchar2(3);
```

Script Output x | Query Result x | Task completed in 0.033 seconds

Table ANGAJATI created.

Table FILM created.

Type PER_DIF compiled

Type LISTA_PERIOADE compiled

Table DIFUZEAZA created.

Doms Output

CREATE+INSERARE_TABLE.sql - Structure No Structure

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\CREATE+INSERARE_TABLE.sql

File Edit View Navigate Run Source Team Tools Window Help

Oracle Connect PROJECT

CREATE+INSERARE_TABLE.sql Ex 6.sql Ex 7.sql Ex 8.sql Ex 9.sql Ex 10.sql Ex 11.sql Ex 12.sql Ex 13.sql

SQL Worksheet History

Worksheet Query Builder

```
CREATE TABLE clienti(
cod_client number(2) constraint pk_client primary key,
nume_client varchar2(30) not null,
prenume_client varchar2(30) not null,
marcute_bonus number(5) );
```

Script Output x | Query Result x | Task completed in 0.033 seconds

Table DIFUZEAZA created.

Table CLIENTI created.

Table RECENZIE created.

Table REZERVARE created.

Doms Output

CREATE+INSERARE_TABLE.sql - Structure No Structure

Click on an identifier with the Control key down to run from "Go to Declaration".

5. Inserare date în tabele

ORAS

```
INSERT INTO oras VALUES (50, 'BV', 'Predeal');
INSERT INTO oras VALUES (51, 'CJ', 'Cluj-Napoca');
INSERT INTO oras VALUES (52, 'GJ', 'Targu Jiu');
INSERT INTO oras VALUES (53, 'IS', 'Pascani');
INSERT INTO oras VALUES (54, 'IS', 'Targu Frumos');
INSERT INTO oras VALUES (55, 'IF', 'Buftea');
INSERT INTO oras VALUES (56, 'IF', 'Chitila');
INSERT INTO oras VALUES (57, 'MS', 'Sighisoara');
INSERT INTO oras VALUES (58, 'TM', 'Timisoara');
INSERT INTO oras VALUES (59, 'VS', 'Vaslui');
```

The screenshot shows the Oracle SQL Developer interface. The top menu bar includes File, Edit, View, Navigate, Run, Source, Team, Tools, Window, and Help. The title bar indicates the file path: E:\FACULTATE\SGDB\PROJECT\CREATE+INSERARE_TABLE.sql. The main workspace contains a Worksheet tab with the following SQL code:

```
INSERT INTO oras VALUES (53, 'IS', 'Pascani');
INSERT INTO oras VALUES (54, 'IS', 'Targu Frumos');
INSERT INTO oras VALUES (55, 'IF', 'Buftea');
INSERT INTO oras VALUES (56, 'IF', 'Chitila');
INSERT INTO oras VALUES (57, 'MS', 'Sighisoara');
INSERT INTO oras VALUES (58, 'TM', 'Timisoara');
INSERT INTO oras VALUES (59, 'VS', 'Vaslui');

select * from oras;
```

Below the worksheet is a Query Result tab displaying the data inserted into the 'oras' table:

COD_ORAS	PREFIJ_JUDET	NUME_ORAS
1	50 BV	Predeal
2	51 CJ	Cluj-Napoca
3	52 GJ	Targu Jiu
4	53 IS	Pascani
5	54 IS	Targu Frumos
6	55 IF	Buftea
7	56 IF	Chitila
8	57 MS	Sighisoara
9	58 TM	Timisoara
10	59 VS	Vaslui

ADRESA

```
INSERT INTO adresa VALUES (10, 50, 'Strada Predeal');
INSERT INTO adresa VALUES (11, 51, 'Strada Unirii');
INSERT INTO adresa VALUES (12, 52, 'Strada Victoriei');
INSERT INTO adresa VALUES (13, 53, 'Strada Garii');
INSERT INTO adresa VALUES (14, 54, 'Strada Horea');
INSERT INTO adresa VALUES (15, 55, 'Strada 1 Decembrie');
INSERT INTO adresa VALUES (16, 56, 'Strada Principala');
INSERT INTO adresa VALUES (17, 57, 'Strada Cetatii');
INSERT INTO adresa VALUES (18, 58, 'Strada Banatului');
INSERT INTO adresa VALUES (19, 59, 'Strada Marasesti');
```

The screenshot shows the Oracle SQL Developer interface. The 'Worksheet' tab is active, containing the following SQL code:

```
CREATE+INSERARE_TABELE.sql Ex 6.sql | Ex 7.sql | Ex 8.sql | Ex 9.sql | Ex 10.sql | Ex 11.sql | Ex 12.sql | Ex 13.sql
SQL Worksheet History PROJECT
Conne... BD PROJECT
Oracle Connect SGRD Oracle NoSQL Co Database Schema
CREATE+INSERARE_TABELE.sql
INSERT INTO adresa VALUES (15, 55, 'Strada 1 Decembrie');
INSERT INTO adresa VALUES (16, 56, 'Strada Principala');
INSERT INTO adresa VALUES (17, 57, 'strada Cetatii');
INSERT INTO adresa VALUES (18, 58, 'Strada Banatului');
INSERT INTO adresa VALUES (19, 59, 'strada Marasesti');

select * from adresa;

INSERT INTO cinematograf VALUES ('C1', 10, 'Cinema Park', 'BV Cinemas', 6);
```

The 'Query Result' tab is active, showing the results of the last query:

COD_ADRESA	COD_ORAS	NUME_STRADA
1	10	50Strada Predeal
2	11	51Strada Unirii
3	12	52Strada Victoriei
4	13	53Strada Garii
5	14	54Strada Horea
6	15	55Strada 1 Decembrie
7	16	56Strada Principala
8	17	57Strada Cetatii
9	18	58Strada Banatului
10	19	59Strada Marasesti

CINEMATOGRAF

```
INSERT INTO cinematograf VALUES ('C1', 10, 'Cinema Park', 'BV Cinemas', 6);
INSERT INTO cinematograf VALUES ('C2', 11, 'Cinema Cotorceni', 'CJ Cinemas', 5);
INSERT INTO cinematograf VALUES ('C3', 12, 'Cinema Victoriei', 'GJ Cinemas', 8);
INSERT INTO cinematograf VALUES ('C4', 13, 'Cinema Garii', 'IS Cinemas', 4);
INSERT INTO cinematograf VALUES ('C5', 14, 'Cinema Luceafarul', 'IS Cinemas', 7);
INSERT INTO cinematograf VALUES ('C6', 15, 'Cinema Grand', 'IF Cinemas', 5);
INSERT INTO cinematograf VALUES ('C7', 16, 'Cinema Aurora', 'IF Cinemas', 6);
INSERT INTO cinematograf VALUES ('C8', 17, 'Cinema Gloria', 'MS Cinemas', 5);
INSERT INTO cinematograf VALUES ('C9', 18, 'Cinema Star', 'TM Cinemas', 5);
INSERT INTO cinematograf VALUES ('C10', 19, 'Cinema Tomis', 'VS Cinemas', 6);
```

The screenshot shows the Oracle SQL Developer interface. The 'Worksheet' tab contains the SQL code for creating the 'cinematograf' table and inserting 10 rows of data. The 'Script Output' tab shows the results of the insertions, displaying 10 rows of data from the 'cinematograf' table. The data is as follows:

	COD_CINEMATOGRAF	COD_ADRESA	NUME_CINEMA	NUME_COMPANIE	NUMAR_SALI
1	C1	10	Cinema Park	BV Cinemas	6
2	C2	11	Cinema Cotorceni	CJ Cinemas	5
3	C3	12	Cinema Victoriei	GJ Cinemas	8
4	C4	13	Cinema Garii	IS Cinemas	4
5	C5	14	Cinema Luceafarul	IS Cinemas	7
6	C6	15	Cinema Grand	IF Cinemas	5
7	C7	16	Cinema Aurora	IF Cinemas	6
8	C8	17	Cinema Gloria	MS Cinemas	5
9	C9	18	Cinema Star	TM Cinemas	5
10	C10	19	Cinema Tomis	VS Cinemas	6

DEPARTAMENTE

```
INSERT INTO departamente VALUES ('D1', 'IT Support');
INSERT INTO departamente VALUES ('D2', 'Marketing');
INSERT INTO departamente VALUES ('D3', 'Resurse Umane');
INSERT INTO departamente VALUES ('D4', 'Logistica');
INSERT INTO departamente VALUES ('D5', 'Supraveghere');
INSERT INTO departamente VALUES ('D6', 'Tehnic');
INSERT INTO departamente VALUES ('D7', 'Vanzari');
INSERT INTO departamente VALUES ('D8', 'Curatenie');
INSERT INTO departamente VALUES ('D9', 'Siguranta si Protectie');
```

The screenshot shows the Oracle SQL Developer interface. The top menu bar includes File, Edit, View, Navigate, Run, Source, Team, Tools, Window, and Help. The title bar indicates the current file is 'CREARE+INSERARE_TABLE.sql'. The left sidebar shows a project structure with 'BD' and 'SGBD' selected. The main workspace contains a 'Worksheet' tab with the following SQL code:

```
INSERT INTO departamente VALUES ('D4', 'Logistica');
INSERT INTO departamente VALUES ('D5', 'Supraveghere');
INSERT INTO departamente VALUES ('D6', 'Tehnic');
INSERT INTO departamente VALUES ('D7', 'Vanzari');
INSERT INTO departamente VALUES ('D8', 'Curatenie');
INSERT INTO departamente VALUES ('D9', 'Siguranta si Protectie');

select * from departamente;
```

Below the worksheet, there are two tabs: 'Script Output' and 'Query Result'. The 'Query Result' tab displays the following data:

COD_DEPARTAMENT	NUME_DEPARTAMENT
1 D1	IT Support
2 D2	Marketing
3 D3	Resurse Umane
4 D4	Logistica
5 D5	Supraveghere
6 D6	Tehnic
7 D7	Vanzari
8 D8	Curatenie
9 D9	Siguranta si Protectie

JOBURI

```
INSERT INTO joburi VALUES ('J1', 'Administrator IT', 4000, 7000, 'Mediu');
INSERT INTO joburi VALUES ('J2', 'Specialist Marketing', 3500, 6000, 'Mediu');
INSERT INTO joburi VALUES ('J3', 'Manager Resurse Umane', 5000, 9000, 'Avansat');
INSERT INTO joburi VALUES ('J4', 'Coordonator Logistica', 4000, 7500, 'Mediu');
INSERT INTO joburi VALUES ('J5', 'Supraveghetor Sali', 2500, 4000, 'Incepator');
INSERT INTO joburi VALUES ('J6', 'Tehnician Echipamente', 3000, 5500, 'Avansat');
INSERT INTO joburi VALUES ('J7', 'Casier', 2000, 3500, 'Incepator');
INSERT INTO joburi VALUES ('J8', 'Personal Curatenie', 1800, 3000, 'Incepator');
INSERT INTO joburi VALUES ('J9', 'Agent de Securitate', 2500, 4500, 'Mediu');
INSERT INTO joburi VALUES ('J10', 'Projectionist', 3000, 5000, 'Mediu');
INSERT INTO joburi VALUES ('J11', 'Coordonator Evenimente', 4000, 6500, 'Mediu');
INSERT INTO joburi VALUES ('J12', 'Consultant Vanzari', 2200, 4000, 'Incepator');
```

The screenshot shows the Oracle SQL Developer interface. The title bar reads "Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\CREATE+INSERARE_TABLE.sql". The menu bar includes File, Edit, View, Navigate, Run, Source, Team, Tools, Window, Help. The toolbar has icons for New Connection, Open Connection, Save, Run, Stop, Refresh, and others. The left sidebar shows a project named "PROJECT" with "BD" and "SGDB" nodes. The main workspace has tabs for "CREATE+INSERARE_TABLE.sql", "Ex 6.sql", "Ex 7.sql", "Ex 8.sql", "Ex 9.sql", "Ex 10.sql", "Ex 11.sql", "Ex 12.sql", and "Ex 13.sql". The "Worksheet" tab is active, displaying the following SQL code:

```
INSERT INTO joburi VALUES ('J12', 'Consultant Vanzari', 2200, 4000, 'Incepator');

select *
from joburi;
```

The "Query Result" tab shows the results of the query, which is a table with columns COD_JOB, NUME_JOB, SALARIU_MINIM, SALARIU_MAXIM, and NIVEL_EXP_NECESAR. The data is as follows:

COD_JOB	NUME_JOB	SALARIU_MINIM	SALARIU_MAXIM	NIVEL_EXP_NECESAR
1 J1	Administrator IT	4000	7000	Mediu
2 J2	Specialist Marketing	3500	6000	Mediu
3 J3	Manager Resurse Umane	5000	9000	Avansat
4 J4	Coordonator Logistica	4000	7500	Mediu
5 J5	Supraveghetor Sali	2500	4000	Incepator
6 J6	Tehnician Echipamente	3000	5500	Avansat
7 J7	Casier	2000	3500	Incepator
8 J8	Personal Curatenie	1800	3000	Incepator
9 J9	Agent de Securitate	2500	4500	Mediu
10 J10	Projectionist	3000	5000	Mediu
11 J11	Coordonator Evenimente	4000	6500	Mediu
12 J12	Consultant Vanzari	2200	4000	Incepator

ANGAJATI

```
INSERT INTO angajati VALUES ('A1', 'J1', 'D1', 'C1','Alexandrescu', 'Andra', 4500, TO_DATE('16-JAN-23', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A2', 'J2', 'D2', 'C1','Roman', 'Cristina', 4500, TO_DATE('20-FEB-23', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A3', 'J3', 'D3', 'C3','Pop', 'Ioana', 5500, TO_DATE('25-MAR-23', 'DD-MON-YY'),'In concediu');
INSERT INTO angajati VALUES ('A4', 'J4', 'D4', 'C4','Georgiu', 'Ana', 4050, TO_DATE('30-APR-23', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A5', 'J5', 'D5', 'C1','Moldovan', 'Roxana', 3200, TO_DATE('02-MAY-23', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A6', 'J6', 'D6', 'C6','David', 'Andreea', 3150, TO_DATE('03-JUN-23', 'DD-MON-YY'),'Suspendat');
INSERT INTO angajati VALUES ('A7', 'J7', 'D7', 'C7','Iacob', 'Andrei', 2300, TO_DATE('18-JUL-19', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A8', 'J1', 'D1', 'C9','Vlonga', 'Stefan', 6000, TO_DATE('20-AUG-23', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A9', 'J2', 'D2', 'C10','Durlesteana', 'Victor', 4000, TO_DATE('25-SEP-23', 'DD-MON-YY'),'In concediu');
INSERT INTO angajati VALUES ('A10', 'J3', 'D3', 'C1','Bechea', 'Flavia', 5000, TO_DATE('24-OCT-19', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A11', 'J4', 'D4', 'C2','Racovita', 'Cristina', 6100, TO_DATE('05-NOV-21', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A12', 'J5', 'D5', 'C3','Igescu', 'Rares', 3200, TO_DATE('13-DEC-23', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A13', 'J6', 'D6', 'C4','Chirila', 'Bianca', 3300, TO_DATE('15-JAN-18', 'DD-MON-YY'),'Suspendat');
INSERT INTO angajati VALUES ('A14', 'J7', 'D7', 'C5','Blaj', 'Deea', 2400, TO_DATE('20-FEB-24', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A15', 'J8', 'D8', 'C6','Buimac', 'Delia', 2800, TO_DATE('25-MAR-20', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A16', 'J9', 'D9', 'C7','Mihalache', 'Diana', 4100, TO_DATE('30-APR-24', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A17', 'J10', 'D6', 'C8','Nechita', 'Teodora', 3600, TO_DATE('05-MAY-24', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A18', 'J4', 'D4', 'C2','Popica', 'Tudor', 4050, TO_DATE('19-JUN-23', 'DD-MON-YY'),'In concediu');
INSERT INTO angajati VALUES ('A19', 'J5', 'D5', 'C10','Florea', 'Bianca', 3300, TO_DATE('15-JUL-20', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A20', 'J11', 'D3', 'C1','Munteanu', 'Andrada', 6400, TO_DATE('27-AUG-22', 'DD-MON-YY'),'Activ');
```

```

INSERT INTO angajati VALUES ('A21', 'J8', 'D8', 'C2','Salcianu', 'Stefan', 2700, TO_DATE('10-SEP-23', 'DD-MON-YY'),'In concediu');
INSERT INTO angajati VALUES ('A22', 'J12', 'D7', 'C3','Monceanu', 'Valentina', 2600, TO_DATE('15-AUG-22', 'DD-MON-YY'),'Suspendat');
INSERT INTO angajati VALUES ('A23', 'J8', 'D8', 'C4','Andruta', 'Andra', 2900, TO_DATE('27-JUL-20', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A24', 'J9', 'D9', 'C9','Tanislav', 'Alexia', 2650, TO_DATE('08-JUN-21', 'DD-MON-YY'),'In concediu');
INSERT INTO angajati VALUES ('A25', 'J10', 'D6', 'C6','Velcea', 'Mihnea', 3400, TO_DATE('01-SEP-23', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A26', 'J10', 'D6', 'C10','Lupu', 'Andrei', 5000, TO_DATE('12-OCT-23', 'DD-MON-YY'),'Suspendat');
INSERT INTO angajati VALUES ('A27', 'J11', 'D3', 'C8','Ciorita', 'Alexandra', 6500, TO_DATE('19-SEP-24', 'DD-MON-YY'),'In concediu');

```

The screenshot shows the Oracle SQL Developer interface with the following details:

- Title Bar:** Oracle SQL Developer E:\VACUUM\SQLDB\PROJECT\CREATE+INSERARE_TABLE.sql
- Toolbar:** File, Edit, View, Navigate, Run, Source, Team, Tools, Window, Help.
- Left Sidebar:** Oracle Connect, PROTECT, Oracle NoSQL, Database Schema.
- Central Area:**
 - Script Output:** Shows the executed SQL code:

```

INSERT INTO angajati VALUES ('A23', 'J8', 'D8', 'C4','Andruta', 'Andra', 2900, TO_DATE('27-JUL-20', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A24', 'J9', 'D9', 'C9','Tanislav', 'Alexia', 2650, TO_DATE('08-JUN-21', 'DD-MON-YY'));
INSERT INTO angajati VALUES ('A25', 'J10', 'D6', 'C6','Velcea', 'Mihnea', 3400, TO_DATE('01-SEP-23', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A26', 'J10', 'D6', 'C10','Lupu', 'Andrei', 5000, TO_DATE('12-OCT-23', 'DD-MON-YY'),'Suspendat');
INSERT INTO angajati VALUES ('A27', 'J11', 'D3', 'C8','Ciorita', 'Alexandra', 6500, TO_DATE('19-SEP-24', 'DD-MON-YY'),'In concediu');

select * from angajati;

```
 - Query Result:** Displays the results of the last query, showing 13 rows of data from the angajati table.
- Bottom Status Bar:** CREARE+INSERIRE_TABLE.sql - Structure, No Structure.

FILM

```
INSERT INTO film VALUES('F1', 'Titanic', 18, 1995, 'Dragoste', 'James Cameron', 'SUA');
INSERT INTO film VALUES('F2', 'Inception', 12, 2010, 'Science Fiction', 'Christopher Nolan', 'SUA');
INSERT INTO film VALUES('F3', 'Parasite', 16, 2019, 'Thriller', 'Bong Joon-ho', 'Coreea de Sud');
INSERT INTO film VALUES('F4', 'The Godfather', 18, 1972, 'Drama', 'Francis Ford Coppola', 'SUA');
INSERT INTO film VALUES('F5', 'Interstellar', 12, 2014, 'Science Fiction', 'Christopher Nolan',
'SUA');
INSERT INTO film VALUES('F6', 'The Shawshank Redemption', 15, 1994, 'Drama', 'Frank
Darabont', 'SUA');
INSERT INTO film VALUES('F7', 'Pulp Fiction', 18, 1994, 'Crima', 'Quentin Tarantino', 'SUA');
INSERT INTO film VALUES('F8', 'La La Land', 12, 2016, 'Musical', 'Damien Chazelle', 'SUA');
INSERT INTO film VALUES('F9', 'Avengers: Endgame', 13, 2019, 'Actiune', 'Anthony Russo', 'SUA');
INSERT INTO film VALUES('F10', 'Joker', 15, 2019, 'Drama', 'Todd Phillips', 'SUA');
INSERT INTO film VALUES('F11', 'The Dark Knight', 13, 2008, 'Supereroi', 'Christopher Nolan',
'SUA');
INSERT INTO film VALUES('F12', 'Spirited Away', 10, 2001, 'Animatie', 'Hayao Miyazaki', 'Japonia');
INSERT INTO film VALUES('F13', 'The Matrix', 16, 1999, 'Science Fiction', 'Lana Wachowski',
'SUA');
INSERT INTO film VALUES('F14', 'Schindler s List', 15, 1993, 'Istoric', 'Steven Spielberg', 'SUA');
INSERT INTO film VALUES('F15', 'Forrest Gump', 12, 1994, 'Drama', 'Robert Zemeckis', 'SUA');
INSERT INTO film VALUES('F16', 'Gladiator', 15, 2000, 'Istoric', 'Ridley Scott', 'SUA');
INSERT INTO film VALUES('F17', 'The Lion King', 6, 1994, 'Animatie', 'Rob Minkoff', 'SUA');
INSERT INTO film VALUES('F18', 'Crouching Tiger, Hidden Dragon', 12, 2000, 'Actiune', 'Ang Lee',
'China');
INSERT INTO film VALUES('F19', 'The Grand Budapest Hotel', 12, 2014, 'Comedie', 'Wes
Anderson', 'SUA');
INSERT INTO film VALUES('F20', 'Black Panther', 13, 2018, 'Supereroi', 'Ryan Coogler', 'SUA');
INSERT INTO film VALUES('F21', 'A Separation', 12, 2011, 'Drama', 'Asghar Farhadi', 'Iran');
INSERT INTO film VALUES('F22', 'Life is Beautiful', 10, 1997, 'Drama', 'Roberto Benigni', 'Italia');
INSERT INTO film VALUES('F23', 'Pan s Labyrinth', 15, 2006, 'Fantzie', 'Guillermo del Toro',
'Spania');
INSERT INTO film VALUES('F24', 'The Revenant', 16, 2015, 'Aventura', 'Alejandro González
Iñárritu', 'SUA');
INSERT INTO film VALUES('F25', 'Amélie', 12, 2001, 'Comedie Romantica', 'Jean-Pierre Jeunet',
'Franta');
```

```

CREATE+INSERARE_TABLE.sql | Ex 6.sql | Ex 7.sql | Ex 8.sql | Ex 9.sql | Ex 10.sql | Ex 11.sql | Ex 12.sql | Ex 13.sql |
SQL Worksheet History | PROJECT |
Worksheet | Query Builder |

```

```

INSERT INTO film VALUES('F23', 'Pan s Labyrinth', 15, 2006, 'Fantzie', 'Guillermo del Toro', 'Spania');
INSERT INTO film VALUES('F24', 'The Revenant', 16, 2015, 'Aventura', 'Alejandro González Iñárritu', 'SUA');
INSERT INTO film VALUES('F25', 'Amélie', 12, 2001, 'Comedie Romantica', 'Jean-Pierre Jeunet', 'Franta');

select *
from film;

```

```

Script Output | Query Result |
SQL | All Rows Fetched: 25 in 0.003 seconds

```

COD_FILM	NUME_FILM	VARSTA_RECOMANDATA	AN_APARITIE	GEN_FILM	NUME_REGIZOR	TARA_PRODUCTIE	
1	F1	Titanic	18	1995	Dragoste	James Cameron	SUA
2	F2	Inception	12	2010	Science Fiction	Christopher Nolan	SUA
3	F3	Parasite	16	2019	Thriller	Bong Joon-ho	Coreea de Sud
4	F4	The Godfather	18	1972	Drama	Francis Ford Coppola	SUA
5	F5	Interstellar	12	2014	Science Fiction	Christopher Nolan	SUA
6	F6	The Shawshank Redemption	15	1994	Drama	Frank Darabont	SUA
7	F7	Pulp Fiction	18	1994	Crima	Quentin Tarantino	SUA
8	F8	La La Land	12	2016	Musical	Damien Chazelle	SUA
9	F9	Avengers: Endgame	13	2019	Action	Anthony Russo	SUA
10	F10	Joker	15	2019	Drama	Todd Phillips	SUA
11	F11	The Dark Knight	13	2008	Supereroi	Christopher Nolan	SUA
12	F12	Spirited Away	10	2001	Animatie	Hayao Miyazaki	Japonia
13	F13	The Matrix	16	1999	Science Fiction	Lana Wachowski	SUA
14	F14	Schindler's List	15	1993	Tatmarie	Steven Spielberg	SUA

CLIENTI

```

INSERT INTO clienti VALUES(1, 'Popescu', 'Ion', 125);
INSERT INTO clienti VALUES(2, 'Ioanitoaiei', 'Maria', 145);
INSERT INTO clienti VALUES(3, 'Georgian', 'Andrei', 190);
INSERT INTO clienti VALUES(4, 'Marin', 'Ana', 110);
INSERT INTO clienti VALUES(5, 'Dumitru', 'Mihai', 175);
INSERT INTO clienti VALUES(6, 'Jalba', 'Elena', 102);
INSERT INTO clienti VALUES(7, 'Popa', 'Florin', 215);
INSERT INTO clienti VALUES(8, 'Stan', 'Ioana', 185);
INSERT INTO clienti VALUES(9, 'Diaconu', 'Cristian', 198);
INSERT INTO clienti VALUES(10, 'Radu', 'Vasile', 135);
INSERT INTO clienti VALUES(11, 'Tudor', 'Monica', 230);
INSERT INTO clienti VALUES(12, 'Dobre', 'Adrian', 140);
INSERT INTO clienti VALUES(13, 'Luca', 'Simona', 155);
INSERT INTO clienti VALUES(14, 'Gheorghe', 'Roxana', 205);
INSERT INTO clienti VALUES(15, 'Nistor', 'Daniel', 180);
INSERT INTO clienti VALUES(16, 'Anghel', 'Alexandra', 145);
INSERT INTO clienti VALUES(17, 'Voicu', 'Alin', 125);
INSERT INTO clienti VALUES(18, 'Petrescu', 'Marian', 192);
INSERT INTO clienti VALUES(19, 'Ciobanu', 'Bianca', 137);
INSERT INTO clienti VALUES(20, 'Gavrilă', 'Robert', 210);

```

```
INSERT INTO clienti VALUES(21, 'Racovita', 'Mihaela', 10);
INSERT INTO clienti VALUES(22, 'Popescu', 'Radu', 230);
```

The screenshot shows the Oracle SQL Developer interface. The 'Worksheet' tab contains two queries:

```
INSERT INTO clienti VALUES(22, 'Popescu', 'Radu', 230);
select * from clienti;
```

The 'Query Result' tab displays the results of the second query, which is a list of 18 rows from the 'clienti' table:

COD_CLIENT	NUME_CLIENT	PRENUME_CLIENT	PUNCTU_BONUS
1	1 Popescu	Ion	125
2	2 Ioanitoaiei	Maria	145
3	3 Georgian	Andrei	190
4	4 Marin	Ana	110
5	5 Dumitru	Mihai	175
6	6 Jalba	Elena	102
7	7 Popa	Florin	215
8	8 Stan	Ioana	185
9	9 Diaconu	Cristian	198
10	10 Radu	Vasile	135
11	11 Tudor	Monica	230
12	12 Dobre	Adrian	140
13	13 Luca	Simona	155
14	14 Gheorghe	Roxana	205
15	15 Nistor	Daniel	180
16	16 Anghel	Alexandra	145
17	17 Voicu	Alin	125
18	18 Patrescu	Marian	102

RECENZIE

```
INSERT INTO recenzie VALUES ('R1', 'F1', 1, 8, TO_DATE('2024-01-15', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R2', 'F2', 2, 9, TO_DATE('2024-06-20', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R3', 'F3', 3, 7, TO_DATE('2024-01-22', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R4', 'F5', 5, 8, TO_DATE('2024-01-28', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R5', 'F8', 2, 7, TO_DATE('2024-10-05', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R6', 'F9', 6, 8, TO_DATE('2024-02-07', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R7', 'F10', 7, 9, TO_DATE('2024-02-10', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R8', 'F11', 8, 10, TO_DATE('2024-12-12', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R9', 'F12', 9, 8, TO_DATE('2024-02-15', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R10', 'F13', 6, 7, TO_DATE('2024-04-18', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R11', 'F15', 9, 9, TO_DATE('2024-12-22', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R12', 'F15', 1, 9, TO_DATE('2024-01-25', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R13', 'F15', 20, 4, TO_DATE('2024-06-28', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R14', 'F16', 11, 3, TO_DATE('2024-09-01', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R15', 'F17', 12, 10, TO_DATE('2024-02-03', 'YYYY-MM-DD'));
```

```

INSERT INTO recenzie VALUES ('R16', 'F18', 13, 4, TO_DATE('2024-03-05', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R17', 'F19', 14, 7, TO_DATE('2024-08-07', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R18', 'F20', 15, 5, TO_DATE('2024-03-10', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R19', 'F20', 15, 7, TO_DATE('2024-08-12', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R20', 'F16', 16, 8, TO_DATE('2024-09-15', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R21', 'F8', 17, 1, TO_DATE('2024-03-18', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R22', 'F10', 18, 1, TO_DATE('2024-12-20', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R23', 'F20', 19, 6, TO_DATE('2024-05-22', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R24', 'F4', 4, 9, TO_DATE('2024-10-15', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R25', 'F15', 15, 2, TO_DATE('2024-04-20', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R26', 'F11', 2, 9, TO_DATE('2024-03-02', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R27', 'F15', 5, 7, TO_DATE('2024-02-10', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R28', 'F17', 5, 8, TO_DATE('2024-01-04', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R29', 'F17', 10, 10, TO_DATE('2024-11-13', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R30', 'F19', 10, 6, TO_DATE('2024-12-16', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R31', 'F23', 1, 7, TO_DATE('2024-12-05', 'YYYY-MM-DD'));

```

The screenshot shows the Oracle SQL Developer interface with the following details:

- Title Bar:** Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\CREATE+INSERARE_TABLE.sql
- Menu Bar:** File, Edit, View, Navigate, Run, Source, Team, Tools, Window, Help
- Toolbar:** Includes icons for New Connection, Open Connection, Save, Run, Stop, Refresh, Undo, Redo, Copy, Paste, Find, Replace, and others.
- Left Sidebar:** Oracle Connect, BD PROJECT, SGBD, Oracle NoSQL DB, Database Schema.
- Central Area:**
 - Worksheet Tab:** Shows the SQL code for creating and inserting data into the 'recenzie' table.
 - Query Result Tab:** Displays the results of the 'select * from recenzie;' query.
 - Script Output Tab:** Shows the execution status: All Rows Fetched: 31 in 0.001 seconds.
- Bottom Status Bar:** CREARE+INSERIRE_TABLE.sql - Structure, No Structure.

SQL Worksheet Content:

```

CREATE+INSERARE_TABLE.sql | Ex 6.sql | Ex 7.sql | Ex 8.sql | Ex 9.sql | Ex 10.sql | Ex 11.sql | Ex 12.sql | Ex 13.sql | 
SQL Worksheet History

INSERT INTO recenzie VALUES ('R30', 'F19', 10, 6, TO_DATE('2024-12-16', 'YYYY-MM-DD'));
INSERT INTO recenzie VALUES ('R31', 'F23', 1, 7, TO_DATE('2024-12-05', 'YYYY-MM-DD'));

select *
from recenzie;

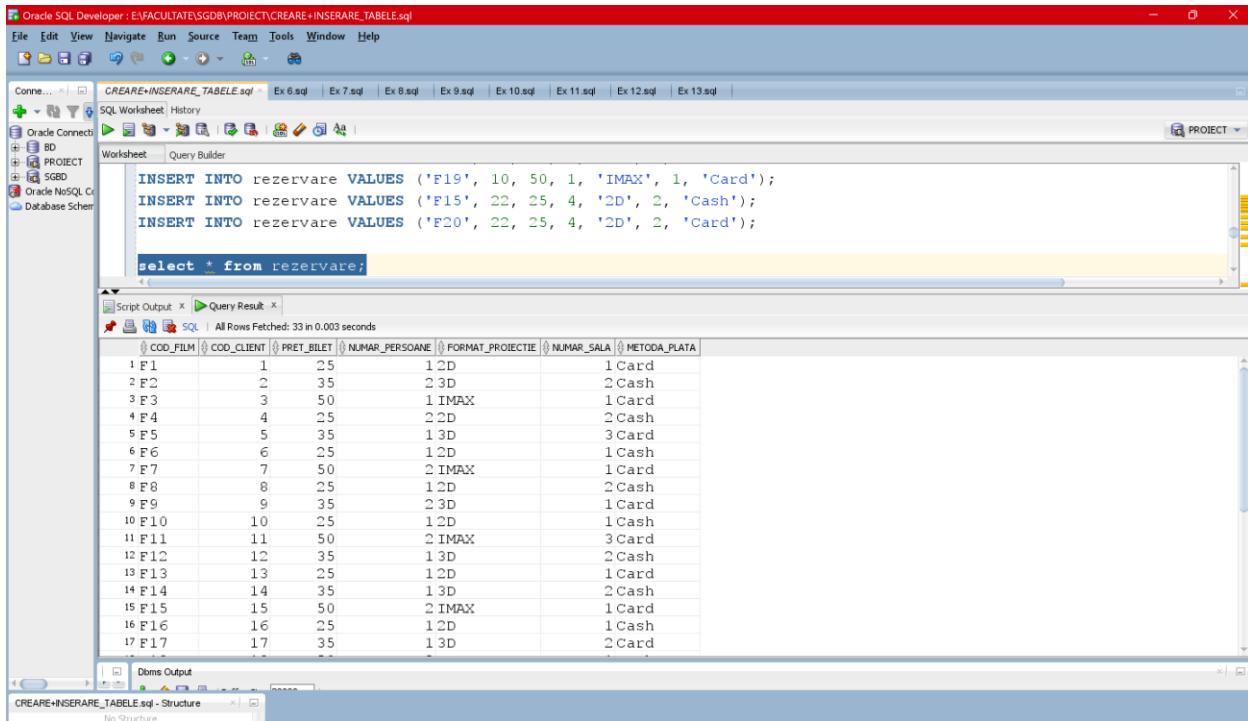
```

Query Result Data:

COD_RECENZIE	COD_FILM	COD_CLIENT	SCOR	DATA_RECENZIE
R.1	F1	1	8	15-JAN-24
R.2	F2	2	9	20-JUN-24
R.3	F3	3	7	22-JAN-24
R.4	F5	5	8	28-JAN-24
R.5	F8	2	7	05-OCT-24
R.6	F9	6	8	07-FEB-24
R.7	F10	7	9	10-FEB-24
R.8	F11	8	10	12-DEC-24
R.9	F12	9	8	15-FEB-24
R.10	F13	6	7	18-APR-24
R.11	F15	9	9	22-DEC-24
R.12	F15	1	9	25-JAN-24
R.13	F15	20	4	28-JUN-24
R.14	F16	11	3	01-SEP-24
R.15	F17	12	10	03-FEB-24
R.16	F18	13	4	05-MAR-24
R.17	F19	14	7	07-AUG-24
R.18	F20	15	5	10-MAR-24

REZERVARE

```
INSERT INTO rezervare VALUES ('F1', 1, 25, 1, '2D', 1, 'Card');
INSERT INTO rezervare VALUES ('F2', 2, 35, 2, '3D', 2, 'Cash');
INSERT INTO rezervare VALUES ('F3', 3, 50, 1, 'IMAX', 1, 'Card');
INSERT INTO rezervare VALUES ('F4', 4, 25, 2, '2D', 2, 'Cash');
INSERT INTO rezervare VALUES ('F5', 5, 35, 1, '3D', 3, 'Card');
INSERT INTO rezervare VALUES ('F6', 6, 25, 1, '2D', 1, 'Cash');
INSERT INTO rezervare VALUES ('F7', 7, 50, 2, 'IMAX', 1, 'Card');
INSERT INTO rezervare VALUES ('F8', 8, 25, 1, '2D', 2, 'Cash');
INSERT INTO rezervare VALUES ('F9', 9, 35, 2, '3D', 1, 'Card');
INSERT INTO rezervare VALUES ('F10', 10, 25, 1, '2D', 1, 'Cash');
INSERT INTO rezervare VALUES ('F11', 11, 50, 2, 'IMAX', 3, 'Card');
INSERT INTO rezervare VALUES ('F12', 12, 35, 1, '3D', 2, 'Cash');
INSERT INTO rezervare VALUES ('F13', 13, 25, 1, '2D', 1, 'Card');
INSERT INTO rezervare VALUES ('F14', 14, 35, 1, '3D', 2, 'Cash');
INSERT INTO rezervare VALUES ('F15', 15, 50, 2, 'IMAX', 1, 'Card');
INSERT INTO rezervare VALUES ('F16', 16, 25, 1, '2D', 1, 'Cash');
INSERT INTO rezervare VALUES ('F17', 17, 35, 1, '3D', 2, 'Card');
INSERT INTO rezervare VALUES ('F18', 18, 50, 2, 'IMAX', 1, 'Cash');
INSERT INTO rezervare VALUES ('F19', 19, 25, 1, '2D', 2, 'Card');
INSERT INTO rezervare VALUES ('F20', 20, 35, 1, '3D', 3, 'Cash');
INSERT INTO rezervare VALUES ('F20', 1, 25, 1, '2D', 2, 'Card');
INSERT INTO rezervare VALUES ('F21', 1, 35, 1, '3D', 3, 'Cash');
INSERT INTO rezervare VALUES ('F23', 1, 50, 1, 'IMAX', 2, 'Card');
INSERT INTO rezervare VALUES ('F23', 21, 35, 3, '3D', 4, 'Card');
INSERT INTO rezervare VALUES ('F10', 2, 25, 1, '2D', 2, 'Cash');
INSERT INTO rezervare VALUES ('F11', 2, 50, 1, 'IMAX', 2, 'Card');
INSERT INTO rezervare VALUES ('F15', 5, 35, 1, '3D', 3, 'Card');
INSERT INTO rezervare VALUES ('F17', 5, 50, 2, 'IMAX', 3, 'Cash');
INSERT INTO rezervare VALUES ('F17', 10, 25, 2, '2D', 2, 'Card');
INSERT INTO rezervare VALUES ('F18', 10, 35, 2, '3D', 3, 'Cash');
INSERT INTO rezervare VALUES ('F19', 10, 50, 1, 'IMAX', 1, 'Card');
INSERT INTO rezervare VALUES ('F15', 22, 25, 4, '2D', 2, 'Cash');
INSERT INTO rezervare VALUES ('F20', 22, 25, 4, '2D', 2, 'Card');
```



DIFUZEAZA

```

INSERT INTO difuzeaza VALUES (
    'C1', 'F1',
    lista_perioade(
        per_dif(TO_DATE('17-06-2024', 'DD-MM-YYYY'), TO_DATE('18:00', 'HH24:MI'),
        TO_DATE('20:00', 'HH24:MI')),
        per_dif(TO_DATE('17-06-2024', 'DD-MM-YYYY'), TO_DATE('15:00', 'HH24:MI'),
        TO_DATE('17:30', 'HH24:MI')),
        per_dif(TO_DATE('18-06-2024', 'DD-MM-YYYY'), TO_DATE('20:00', 'HH24:MI'),
        TO_DATE('22:30', 'HH24:MI'))),
    'Romana',150);

```

```

INSERT INTO difuzeaza VALUES (
    'C1', 'F2',
    lista_perioade(
        per_dif(TO_DATE('05-12-2024', 'DD-MM-YYYY'), TO_DATE('18:00', 'HH24:MI'),
        TO_DATE('20:30', 'HH24:MI')),
        per_dif(TO_DATE('05-12-2024', 'DD-MM-YYYY'), TO_DATE('21:00', 'HH24:MI'),
        TO_DATE('23:30', 'HH24:MI'))),
    'Romana', 180);

```

```
INSERT INTO difuzeaza VALUES (
    'C1', 'F2',
    lista_perioade(
        per_dif(TO_DATE('05-12-2024', 'DD-MM-YYYY'), TO_DATE('18:00', 'HH24:MI'),
        TO_DATE('20:30', 'HH24:MI')),
        per_dif(TO_DATE('05-12-2024', 'DD-MM-YYYY'), TO_DATE('21:00', 'HH24:MI'),
        TO_DATE('23:00', 'HH24:MI'))),
    'Romana', 150);
```

```
INSERT INTO difuzeaza VALUES (
    'C2', 'F3',
    lista_perioade(
        per_dif(TO_DATE('10-12-2024', 'DD-MM-YYYY'), TO_DATE('18:00', 'HH24:MI'),
        TO_DATE('20:00', 'HH24:MI')),
        per_dif(TO_DATE('12-12-2024', 'DD-MM-YYYY'), TO_DATE('16:00', 'HH24:MI'),
        TO_DATE('18:30', 'HH24:MI')),
        per_dif(TO_DATE('15-12-2024', 'DD-MM-YYYY'), TO_DATE('20:00', 'HH24:MI'),
        TO_DATE('22:30', 'HH24:MI'))),
    'Engleza', 150);
```

```
INSERT INTO difuzeaza VALUES (
    'C2', 'F4',
    lista_perioade(
        per_dif(TO_DATE('15-12-2024', 'DD-MM-YYYY'), TO_DATE('18:00', 'HH24:MI'),
        TO_DATE('20:00', 'HH24:MI')),
        per_dif(TO_DATE('16-12-2024', 'DD-MM-YYYY'), TO_DATE('14:00', 'HH24:MI'),
        TO_DATE('16:00', 'HH24:MI'))),
    'Romana', 120);
```

```
INSERT INTO difuzeaza VALUES (
    'C3', 'F5',
    lista_perioade(
        per_dif(TO_DATE('10-12-2024', 'DD-MM-YYYY'), TO_DATE('18:00', 'HH24:MI'),
        TO_DATE('20:30', 'HH24:MI')),
        per_dif(TO_DATE('03-12-2024', 'DD-MM-YYYY'), TO_DATE('17:00', 'HH24:MI'),
        TO_DATE('19:00', 'HH24:MI'))),
    'Romana', 150);
```

```
INSERT INTO difuzeaza VALUES (
    'C4', 'F6',
    lista_perioade(
        per_dif(TO_DATE('20-12-2024', 'DD-MM-YYYY'), TO_DATE('17:00', 'HH24:MI'),
        TO_DATE('19:00', 'HH24:MI'))),
```

```
per_dif(TO_DATE('21-12-2024', 'DD-MM-YYYY'), TO_DATE('18:00', 'HH24:MI'),
TO_DATE('20:00', 'HH24:MI'))),
'Engleza', 120);
```

```
INSERT INTO difuzeaza VALUES (
'C5', 'F7',
lista_perioade(
per_dif(TO_DATE('09-12-2024', 'DD-MM-YYYY'), TO_DATE('16:00', 'HH24:MI'),
TO_DATE('18:00', 'HH24:MI')),
per_dif(TO_DATE('02-12-2024', 'DD-MM-YYYY'), TO_DATE('19:00', 'HH24:MI'),
TO_DATE('21:00', 'HH24:MI'))),
'Romana', 120);
```

```
INSERT INTO difuzeaza VALUES (
'C6', 'F8',
lista_perioade(
per_dif(TO_DATE('05-12-2024', 'DD-MM-YYYY'), TO_DATE('18:00', 'HH24:MI'),
TO_DATE('21:00', 'HH24:MI')),
per_dif(TO_DATE('06-12-2024', 'DD-MM-YYYY'), TO_DATE('17:30', 'HH24:MI'),
TO_DATE('20:00', 'HH24:MI'))),
'Romana', 150);
```

```
INSERT INTO difuzeaza VALUES (
'C7', 'F9',
lista_perioade(
per_dif(TO_DATE('04-12-2024', 'DD-MM-YYYY'), TO_DATE('15:00', 'HH24:MI'),
TO_DATE('17:30', 'HH24:MI')),
per_dif(TO_DATE('04-12-2024', 'DD-MM-YYYY'), TO_DATE('18:00', 'HH24:MI'),
TO_DATE('20:30', 'HH24:MI'))),
'Romana', 150);
```

```
INSERT INTO difuzeaza VALUES (
'C8', 'F10',
lista_perioade(
per_dif(TO_DATE('15-12-2024', 'DD-MM-YYYY'), TO_DATE('14:00', 'HH24:MI'),
TO_DATE('15:30', 'HH24:MI')),
per_dif(TO_DATE('16-12-2024', 'DD-MM-YYYY'), TO_DATE('16:00', 'HH24:MI'),
TO_DATE('17:30', 'HH24:MI'))),
'Engleza', 90);
```

```
INSERT INTO difuzeaza VALUES (
'C10', 'F12',
lista_perioade(
```

```
    per_dif(TO_DATE('05-12-2024', 'DD-MM-YYYY'), TO_DATE('18:00', 'HH24:MI'),
TO_DATE('19:30', 'HH24:MI')),
    per_dif(TO_DATE('02-12-2024', 'DD-MM-YYYY'), TO_DATE('20:00', 'HH24:MI'),
TO_DATE('22:00', 'HH24:MI'))),
    'Japoneza', 90);
```

```
INSERT INTO difuzeaza VALUES (
'C1', 'F13',
lista_perioade(
    per_dif(TO_DATE('10-12-2024', 'DD-MM-YYYY'), TO_DATE('17:30', 'HH24:MI'),
TO_DATE('20:00', 'HH24:MI')),
    per_dif(TO_DATE('11-12-2024', 'DD-MM-YYYY'), TO_DATE('16:00', 'HH24:MI'),
TO_DATE('18:30', 'HH24:MI'))),
    'Romana', 150);
```

```
INSERT INTO difuzeaza VALUES (
'C2', 'F14',
lista_perioade(
    per_dif(TO_DATE('15-12-2024', 'DD-MM-YYYY'), TO_DATE('16:00', 'HH24:MI'),
TO_DATE('18:00', 'HH24:MI'))),
    'Romana', 120);
```

```
INSERT INTO difuzeaza VALUES (
'C3', 'F15',
lista_perioade(
    per_dif(TO_DATE('12-12-2024', 'DD-MM-YYYY'), TO_DATE('19:00', 'HH24:MI'),
TO_DATE('21:00', 'HH24:MI'))),
    'Romana', 120);
```

```
INSERT INTO difuzeaza VALUES (
'C4', 'F16',
lista_perioade(
    per_dif(TO_DATE('20-12-2024', 'DD-MM-YYYY'), TO_DATE('16:00', 'HH24:MI'),
TO_DATE('18:30', 'HH24:MI')),
    per_dif(TO_DATE('21-12-2024', 'DD-MM-YYYY'), TO_DATE('15:00', 'HH24:MI'),
TO_DATE('17:30', 'HH24:MI')),
    per_dif(TO_DATE('22-12-2024', 'DD-MM-YYYY'), TO_DATE('18:00', 'HH24:MI'),
TO_DATE('20:30', 'HH24:MI'))),
    'Engleza', 150);
```

```
INSERT INTO difuzeaza VALUES (
'C5', 'F17',
lista_perioade(
```

```

    per_dif(TO_DATE('18-12-2024', 'DD-MM-YYYY'), TO_DATE('15:00', 'HH24:MI'),
TO_DATE('16:20', 'HH24:MI')),
    per_dif(TO_DATE('02-12-2024', 'DD-MM-YYYY'), TO_DATE('14:30', 'HH24:MI'),
TO_DATE('15:50', 'HH24:MI'))),
    'Romana', 80);

```

```

INSERT INTO difuzeaza VALUES (
    'C6', 'F18',
    lista_perioade(
        per_dif(TO_DATE('10-12-2024', 'DD-MM-YYYY'), TO_DATE('17:00', 'HH24:MI'),
TO_DATE('18:30', 'HH24:MI')),
        per_dif(TO_DATE('12-12-2024', 'DD-MM-YYYY'), TO_DATE('19:00', 'HH24:MI'),
TO_DATE('20:30', 'HH24:MI'))),
    'Chineza', 90);

```

```

INSERT INTO difuzeaza VALUES (
    'C7', 'F19',
    lista_perioade(
        per_dif(TO_DATE('05-12-2024', 'DD-MM-YYYY'), TO_DATE('15:00', 'HH24:MI'),
TO_DATE('17:10', 'HH24:MI')),
        per_dif(TO_DATE('06-12-2024', 'DD-MM-YYYY'), TO_DATE('18:00', 'HH24:MI'),
TO_DATE('20:10', 'HH24:MI')),
        per_dif(TO_DATE('07-12-2024', 'DD-MM-YYYY'), TO_DATE('17:30', 'HH24:MI'),
TO_DATE('19:40', 'HH24:MI'))),
    'Romana', 130);

```

```

INSERT INTO difuzeaza VALUES (
    'C8', 'F20',
    lista_perioade(
        per_dif(TO_DATE('19-12-2024', 'DD-MM-YYYY'), TO_DATE('16:00', 'HH24:MI'),
TO_DATE('18:20', 'HH24:MI')),
        per_dif(TO_DATE('02-12-2024', 'DD-MM-YYYY'), TO_DATE('18:30', 'HH24:MI'),
TO_DATE('20:50', 'HH24:MI'))),
    'Engleza', 140);

```

```

INSERT INTO difuzeaza VALUES (
    'C9', 'F21',
    lista_perioade(
        per_dif(TO_DATE('10-12-2024', 'DD-MM-YYYY'), TO_DATE('18:00', 'HH24:MI'),
TO_DATE('20:00', 'HH24:MI'))),
    'Romana', 120);

```

```
INSERT INTO difuzeaza VALUES (
    'C10', 'F22',
    lista_perioade(
        per_dif(TO_DATE('23-12-2024', 'DD-MM-YYYY'), TO_DATE('17:00', 'HH24:MI'),
        TO_DATE('18:10', 'HH24:MI'))),
    'Italiana', 70);
```

```
INSERT INTO difuzeaza VALUES (
    'C1', 'F23',
    lista_perioade(
        per_dif(TO_DATE('15-12-2024', 'DD-MM-YYYY'), TO_DATE('19:00', 'HH24:MI'),
        TO_DATE('20:50', 'HH24:MI'))),
    'Spaniola', 110);
```

```
INSERT INTO difuzeaza VALUES (
    'C2', 'F24',
    lista_perioade(
        per_dif(TO_DATE('27-12-2024', 'DD-MM-YYYY'), TO_DATE('16:00', 'HH24:MI'),
        TO_DATE('17:30', 'HH24:MI')),
        per_dif(TO_DATE('02-12-2024', 'DD-MM-YYYY'), TO_DATE('18:00', 'HH24:MI'),
        TO_DATE('19:30', 'HH24:MI'))),
    'Romana', 90);
```

```
INSERT INTO difuzeaza VALUES (
    'C3', 'F25',
    lista_perioade(
        per_dif(TO_DATE('10-12-2024', 'DD-MM-YYYY'), TO_DATE('15:30', 'HH24:MI'),
        TO_DATE('17:10', 'HH24:MI'))),
    'Franceza', 100);
```

--afisarea tabelului DIFUZEAZA

The screenshot shows the Oracle SQL Developer interface. In the 'Worksheet' tab, there is a query builder window containing the following SQL code:

```

CREATE+INSERARE_TABLE.sql | Ex 6.sql | Ex 7.sql | Ex 8.sql | Ex 9.sql | Ex 10.sql | Ex 11.sql | Ex 12.sql | Ex 13.sql | PROJECT

INSERT INTO difuzeaza VALUES (
    'C3', 'F25',
    lista_perioade(
        per_dif(TO_DATE('10-12-2024', 'DD-MM-YYYY'), TO_DATE('15:30', 'HH24:MI'), TO_DATE('17:10', 'HH24:MI'))),
    'Franceza', 100);

select *
from difuzeaza;

```

Below the code, the 'Query Result' tab displays the data inserted into the table:

COD_CINEMATOGRAF	COD_FILM	SUBTITRARI	DURATA_FILM
1 C1	F1	SYSTEM.LISTA PERIOADE([SYSTEM.PER.DIF], [SYSTEM.PER.DIF], [SYSTEM.PER.DIF])	Romana 150
2 C1	F2	SYSTEM.LISTA PERIOADE([SYSTEM.PER.DIF], [SYSTEM.PER.DIF])	Romana 180
3 C2	F3	SYSTEM.LISTA PERIOADE([SYSTEM.PER.DIF], [SYSTEM.PER.DIF], [SYSTEM.PER.DIF])	Engleza 150
4 C2	F4	SYSTEM.LISTA PERIOADE([SYSTEM.PER.DIF], [SYSTEM.PER.DIF])	Romana 120
5 C3	F5	SYSTEM.LISTA PERIOADE([SYSTEM.PER.DIF], [SYSTEM.PER.DIF])	Romana 150
6 C4	F6	SYSTEM.LISTA PERIOADE([SYSTEM.PER.DIF], [SYSTEM.PER.DIF])	Engleza 120
7 C5	F7	SYSTEM.LISTA PERIOADE([SYSTEM.PER.DIF], [SYSTEM.PER.DIF])	Romana 120
8 C6	F8	SYSTEM.LISTA PERIOADE([SYSTEM.PER.DIF], [SYSTEM.PER.DIF])	Romana 150
9 C7	F9	SYSTEM.LISTA PERIOADE([SYSTEM.PER.DIF], [SYSTEM.PER.DIF])	Romana 150
10 C8	F10	SYSTEM.LISTA PERIOADE([SYSTEM.PER.DIF], [SYSTEM.PER.DIF])	Engleza 90
11 C10	F12	SYSTEM.LISTA PERIOADE([SYSTEM.PER.DIF], [SYSTEM.PER.DIF])	Japoneza 90
12 C1	F13	SYSTEM.LISTA PERIOADE([SYSTEM.PER.DIF], [SYSTEM.PER.DIF])	Romana 150
13 C2	F14	SYSTEM.LISTA PERIOADE([SYSTEM.PER.DIF])	Romana 120
14 C3	F15	SYSTEM.LISTA PERIOADE([SYSTEM.PER.DIF])	Romana 120

--afisarea tabloului imbricat perioade_difuzare din tabelul DIFUZEAZA

The screenshot shows the Oracle SQL Developer interface. In the 'Worksheet' tab, there is a query builder window containing the following SQL code:

```

CREATE+INSERARE_TABLE.sql | Ex 6.sql | Ex 7.sql | Ex 8.sql | Ex 9.sql | Ex 10.sql | Ex 11.sql | Ex 12.sql | Ex 13.sql | PROJECT

SELECT d.cod_cinematograf,
       d.cod_film,
       TO_CHAR(p.data_difuzare, 'DD-MM-YYYY') AS data_difuzare,
       TO_CHAR(p.ora_inceput, 'HH24:MI') AS ora_inceput,
       TO_CHAR(p.ora_final, 'HH24:MI') AS ora_final
  FROM difuzeaza d,
       TABLE(d.perioade_difuzare) p;

```

Below the code, the 'Query Result' tab displays the data from the nested table:

COD_CINEMATOGRAF	COD_FILM	DATA_DIFUZARE	ORA_INCEPUT	ORA_FINAL
1 C1	F1	17-06-2024 18:00	20:00	
2 C1	F1	17-06-2024 15:00	17:30	
3 C1	F1	18-06-2024 20:00	22:30	
4 C1	F2	05-12-2024 18:00	20:30	
5 C1	F2	05-12-2024 21:00	23:30	
6 C2	F3	10-12-2024 18:00	20:00	
7 C2	F3	12-12-2024 16:00	18:30	
8 C2	F3	15-12-2024 20:00	22:30	
9 C2	F4	15-12-2024 18:00	20:00	
10 C2	F4	16-12-2024 14:00	16:00	
11 C3	F5	10-12-2024 18:00	20:30	
12 C3	F5	03-12-2024 17:00	19:00	
13 C4	F6	08-12-2024 17:00	19:00	

6. Procedură care utilizează cele 3 tipuri de colecții studiate

Se dorește realizarea unei analize pentru identificarea, în fiecare cinematograf, a filmelor care sunt difuzate în prezent și a clientilor care au rezervări active pentru aceste filme. În plus, se vor extrage recenziile oferite de clienți pentru filmele respective, dar doar dacă rezervările lor sunt active. Recenziile asociate vizionărilor anterioare nu sunt relevante în această cerință, care se concentrează exclusiv pe situația rezervărilor active.

- Se folosesc tablouri indexate pentru a stoca informații despre cinematografe și filme.
 - Se folosesc tablouri imbicate pentru a stoca informațiile despre rezervările și clienții.
 - Se folosesc vectori pentru a gestiona recenziile acordate de către clienți.
- Tabele utilizate: cinematograf, film, clienti, rezervare, difuzeaza, recenzii

```
create or replace procedure ex6
is
    -- Definire tipuri de date pentru afisarea informațiilor despre filme
    type tablou_indexat_nume is table of film.nume_film%type index by pls_integer;
    type tablou_indexat_cod is table of film.cod_film%type index by pls_integer;
    t_indexat_filme_nume tablou_indexat_nume;
    t_indexat_filme_cod tablou_indexat_cod;

    cursor c_cinematograf is
        select cod_cinematograf as cod, nume_cinema as nume
        from cinematograf;

    cursor c_film(cod_cinematograf.cod_cinematograf%type) is
        select f.nume_film as nume, f.cod_film as cod
        from difuzeaza d
        join film f on d.cod_film = f.cod_film
        where d.cod_cinematograf = cod;

    -- Definire tipuri de date pentru afisarea clientilor
    type tablou_imbricat1 is table of clienti.nume_client%type;
    type tablou_imbricat2 is table of clienti.cod_client%type;

    cursor c_client(cod_film.cod_film%type) is
        select distinct c.nume_client, c.cod_client
        from clienti c
```

```

join rezervare r on c.cod_client = r.cod_client
where r.cod_film = cod;

-- Definire vector pentru afisarea recenziilor
type vector is varray(20) of recenzie.scor%type;

cursor c_recenzie(cod_film.cod_film%type) is
select rec.scor, rec.cod_client
from recenzie rec
join clienti c on rec.cod_client = c.cod_client
join film f on rec.cod_film = f.cod_film
where f.cod_film = cod
-- blocul EXISTS este folosit pentru a valida ca recenzia selectata apartine unui
-- client care a facut o rezervare pentru filmul respectiv
and exists (
    select 1
    from rezervare rez
    where rez.cod_film = f.cod_film
        and rez.cod_client = rec.cod_client
);
-- fara existenta blocului EXISTS s-ar returna toate recenziile inregistrate vreodata
-- pentru filmul respectiv

begin
-- Se parcurge fiecare cinematograf
for i in c_cinematograf loop
    dbms_output.put_line('-----');
    dbms_output.put_line('Cinematograf: ' || i.nume);
    dbms_output.put_line('-----');

    open c_film(i.cod);
    fetch c_film bulk collect into t_indexat_filme_nume, t_indexat_filme_cod;

    -- Se afiseaza filmele care corespund cinematografului curent
    if t_indexat_filme_nume.count = 0 then
        dbms_output.put_line('Nu exista filme difuzate in acest cinematograf.');
    else
        for j in t_indexat_filme_nume.first..t_indexat_filme_nume.last loop
            dbms_output.put_line(' Film: ' || t_indexat_filme_nume(j));

            -- Se afiseaza clientii care au rezervari pentru filmul curent
            open c_client(t_indexat_filme_cod(j));
            declare
                -- Se va initializa pentru fiecare film un vector cu recenziile lasate de clienti

```

```

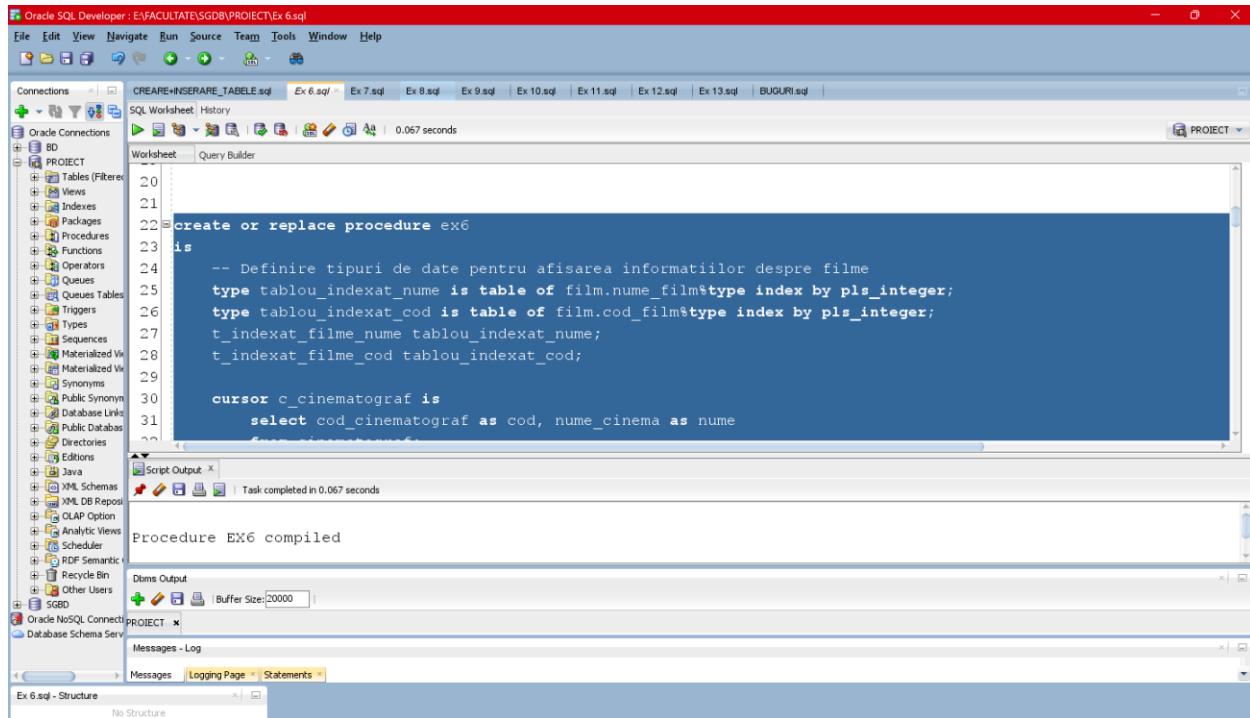
v_recenzi vector := vector();
t_imbricat1_clienti tablou_imbricat1 := tablou_imbricat1(); -- se retin numele
clientilor
    t_imbricat2_clienti tablou_imbricat2 := tablou_imbricat2(); -- codurile clientilor care
au facut rezervari
    t_imbricat3_clienti tablou_imbricat2 := tablou_imbricat2(); -- codurile clientilor care
au lasat recenzie
begin
    fetch c_client bulk collect into t_imbricat1_clienti, t_imbricat2_clienti;
    if t_imbricat1_clienti.count = 0 then
        dbms_output.put_line(' Nu exista clienti cu rezervari pentru acest film.');
    else
        dbms_output.put(' Clienti cu rezervari: ');
        for k in t_imbricat1_clienti.first..t_imbricat1_clienti.last loop
            dbms_output.put(t_imbricat1_clienti(k));
            if k < t_imbricat1_clienti.last then
                dbms_output.put(',');
            end if;
        end loop;
        dbms_output.new_line;

-- Se afiseaza recenziile pentru filmul curent
open c_recenzie(t_indexat_filme_cod(j));
fetch c_recenzie bulk collect into v_recenzi, t_imbricat3_clienti;

if v_recenzi.count = 0 then
    dbms_output.put_line(' Nu exista recenzie pentru acest film.');
else
    for k in t_imbricat1_clienti.first..t_imbricat1_clienti.last loop
        -- bloc necesar pentru a determina daca clientul curent a acordat sau nu
recenzie filmului curent
        declare
            -- cu aceasta variabila se verifica existenta unei recenzie date de clientul
curent
            nota_existenta boolean := false;
        begin
            for idx in 1..v_recenzi.count loop
                -- daca clientul care a facut o rezervare la filmul curent
                -- a acordat si o recenzie, atunci se va afisa nota lui
                if t_imbricat2_clienti(k) = t_imbricat3_clienti(idx) then
                    dbms_output.put_line(' ' || t_imbricat1_clienti(k) || ' a acordat nota:
' || v_recenzi(idx));
                    nota_existenta := true;
                    exit;
                end if;
            end loop;
        end;
    end loop;
end;

```

```
        end if;
    end loop;
    if not nota_existenta then
        dbms_output.put_line('' || t_imbricat1_clienti(k) || ' nu a acordat
nicio nota.');
        end if;
    end;
    end loop;
end if;
close c_recenzie;
end if;
end;
close c_client;
end loop;
end if;
dbms_output.new_line;
close c_film;
end loop;
end ex6;
/
```



Testare procedură ex6

EXECUTE EX6;

The screenshot shows the Oracle SQL Developer interface. In the central workspace, the query 'EXECUTE EX6;' is run, resulting in the message 'PL/SQL procedure successfully completed.' Below this, the output window displays results for 'Cinematograf: Cinema Cotroceni'. The results show film names and associated client information. Two specific lines are highlighted with yellow arrows: 'Nu exista recenzii pentru acest film.' and 'Nu exista clienti cu rezervari pentru acest film.'.

```
CREATE+INSERARE_TABLELE.sql Ex 6.sql Ex 7.sql Ex 8.sql Ex 9.sql Ex 10.sql Ex 11.sql Ex 12.sql Ex 13.sql BUGURI.sql Welcome Page PROJECT

EXECUTE EX6;

PL/SQL procedure successfully completed.

Doms Output
PROJECT x
-----Cinematograf: Cinema Cotroceni-----
Film: Parasite
Clienti cu rezervari: Georgian
Georgian a acordat nota: 7
Film: The Godfather
Clienti cu rezervari: Marin
Marin a acordat nota: 9
Film: Schindler's List
Clienti cu rezervari: Gheorghe
Nu exista recenzii pentru acest film. ←
Film: The Revenant
Nu exista clienti cu rezervari pentru acest film. ←

Compiler - Log
```

The screenshot shows the Oracle SQL Developer interface. In the central workspace, the query 'EXECUTE EX6;' is run, resulting in the message 'PL/SQL procedure successfully completed.' Below this, the output window displays results for 'Cinematograf: Cinema Victoriei'. The results show film names and associated client information. One specific line is highlighted with a yellow arrow: 'Popescu nu a acordat nicio nota.'.

```
CREATE+INSERARE_TABLELE.sql Ex 6.sql Ex 7.sql Ex 8.sql Ex 9.sql Ex 10.sql Ex 11.sql Ex 12.sql Ex 13.sql BUGURI.sql Welcome Page PROJECT

EXECUTE EX6;

PL/SQL procedure successfully completed.

Doms Output
PROJECT x
-----Cinematograf: Cinema Victoriei-----
Film: Interstellar
Clienti cu rezervari: Dumitru
Dumitru a acordat nota: 8
Film: Forrest Gump
Clienti cu rezervari: Dumitru, Popescu, Nistor
Dumitru a acordat nota: 7
Popescu nu a acordat nicio nota. ←
Nistor a acordat nota: 2
Film: Amélie
Nu exista clienti cu rezervari pentru acest film.
```

7. Procedură care utilizează cursor dinamic

În baza de date a unui cinematograf, pentru fiecare film este cunoscut genul acestuia (de exemplu, Science Fiction, Dramă, Dragoste etc.). Cerința presupune să se primească un gen de film specific, iar pentru toate filmele care aparțin acestui gen să se afișeze următoarele informații:

1. *Perioada de difuzare a fiecărui film.*
2. *Recenziile asociate fiecărui film, constând în data recenziei și scorul acesteia.*

Dacă un film nu are nicio recenzie, trebuie gestionat explicit acest caz, iar mesajul corespunzător să fie afișat, indicând lipsa recenziilor pentru acel film.

-- Tabele utilizate: difuzeaza, film, recenzie

```
create or replace procedure ex7(gen in film.gen_film%type)
is
```

```
    verificare film.cod_film%type; -- variabila pentru a testa existenta genului
    ct_recenzii number; -- variabila pentru a verifica exista recenziilor unui film
```

→ -- cursor pentru filme
cursor c_film is
 select cod_film, nume_film
 from film
 where gen_film = gen;

→ -- cursor parametrizat, dependent de cursorul c_film
cursor c_recenzie(cod_film.cod_film%type) is
 select scor, data_recenzie
 from recenzie
 where cod_film = cod
 order by data_recenzie desc ;

→ -- cursor dinamic pentru perioadele de difuzare
type difuzare_tip is ref cursor;
c_difuzare difuzare_tip;

```
interrogare_sql varchar2(1000);
v_data_difuzare date;
v_ora_inceput date;
v_ora_final date;
```

```

begin
    -- tratarea erorii ca nu exista genul introdus de utilizator
    select cod_film into verificare
    from film
    where gen_film = gen
        and rownum = 1; -- se limiteaza rezultatul cautarii a.i sa returneze doar primul rand,
                        -- daca exista, care indeplineste conditia gen_film = gen

    for i in c_film loop
        ct_recenzii := 0;
        dbms_output.put_line('Numele filmului: ' || i.nume_film || ', iar codul: ' || i.cod_film);
        dbms_output.put_line(' Perioada de difuzare a acestui film este: ');

        interogare_sql := 'select data_difuzare, ora_inceput, ora_final ' ||
                           'from table (select perioade_difuzare ' ||
                           'from difuzeaza ' ||
                           'where cod_film = :cod_film_param)';

        -- se deschide cursorul dinamic
        open c_difuzare for interogare_sql using i.cod_film;
        loop
            fetch c_difuzare into v_data_difuzare, v_ora_inceput, v_ora_final;
            exit when c_difuzare%notfound;

            dbms_output.put_line('    Data: ' || v_data_difuzare || ', intre orele: ' ||
                                 to_char(v_ora_inceput, 'hh24:mi') || '-' || to_char(v_ora_final, 'hh24:mi'));
        end loop;
        close c_difuzare;

        for j in c_recenzie(i.cod_film) loop
            dbms_output.put_line(' Recenzia filmului a fost postata la data: ' || j.data_recenzie || ', iar
nota acordata este: ' || j.scor);
            ct_recenzii := ct_recenzii + 1;
        end loop;

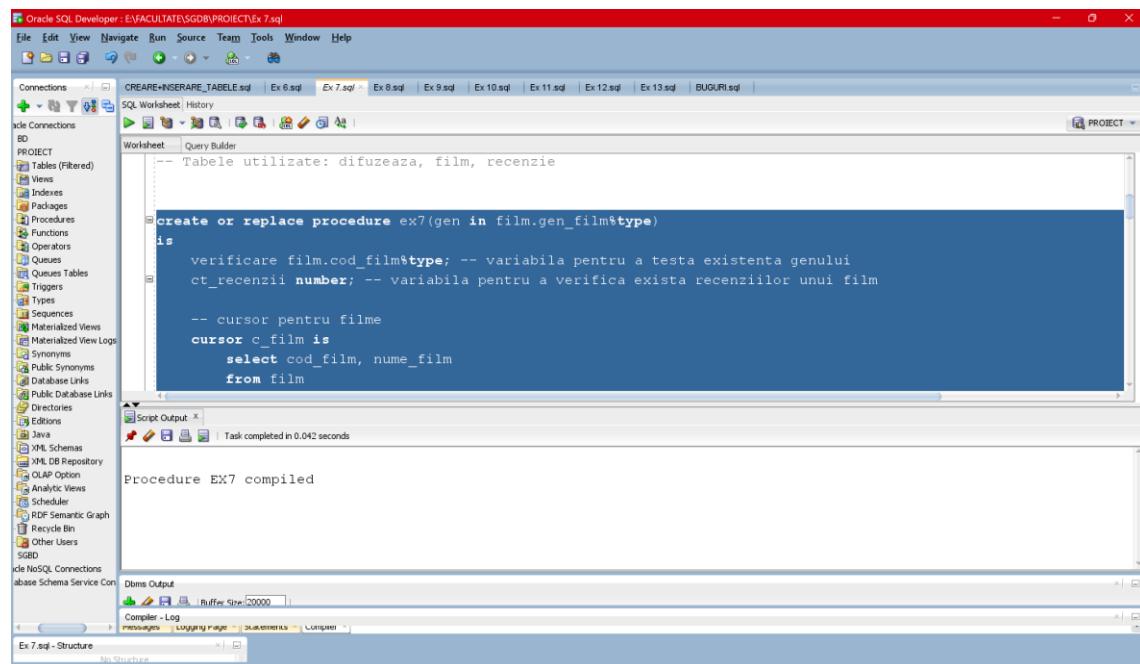
        if ct_recenzii = 0 then
            dbms_output.put_line(' Acest film nu are recenzii.');
        end if;

        dbms_output.new_line;
    end loop;

```

EXCEPTION

```
WHEN NO_DATA_FOUND THEN
    RAISE_APPLICATION_ERROR(-20000, 'Nu exista filme cu genul specificat in baza de date.');
WHEN OTHERS THEN
    RAISE_APPLICATION_ERROR(-20012,'A aparut o eroare: ' || sqlerrm);
end ex7;
/
```



Testare procedura ex7

```
begin
    ex7('Drama');
end;
/
```

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 7.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections PROJECT

SQL Worksheet: History

Worksheet Query Builder

```
begin
    ex7('Drama');
end;
```

Script Output X | Task completed in 0.019 seconds
PL/SQL procedure successfully completed.

Dims Output

PROJECT x

Numele filmului: The Godfather, iar codul: F4
Perioada de difuzare a acestui film este:
Data: 15-DEC-24, intre orele: 18:00-20:00
Data: 16-DEC-24, intre orele: 14:00-16:00
Recenzie filmului a fost postata la data: 15-OCT-24, iar nota acordata este: 9

Numele filmului: The Shawshank Redemption, iar codul: F6
Perioada de difuzare a acestui film este:
Data: 20-DEC-24, intre orele: 17:00-19:00
Data: 21-DEC-24, intre orele: 18:00-20:00

Compiler - Log

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 7.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections PROJECT

SQL Worksheet: History

Worksheet Query Builder

```
begin
    ex7('Drama');
end;
```

Script Output X | Task completed in 0.018 seconds
PL/SQL procedure successfully completed.

Dims Output

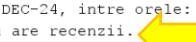
PROJECT x

RECENZIA FILMULUI A FOST POSTATA LA DATA: 20-JAN-24, IAR NOTA ACORDATA ESTE: 9

Numele filmului: A Separation, iar codul: F21
Perioada de difuzare a acestui film este:
Data: 10-DEC-24, intre orele: 18:00-20:00
Acest film nu are recenzii.

Numele filmului: Life is Beautiful, iar codul: F22
Perioada de difuzare a acestui film este:
Data: 23-DEC-24, intre orele: 17:00-18:10
Acest film nu are recenzii.

Compiler - Log



```
execute ex7('Horror');
```

The screenshot shows the Oracle SQL Developer interface. The main window displays a SQL Worksheet with the following code:

```
begin
    ex7('Drama');
end;
/
execute ex7('Horror');
execute ex7('Dragoste');
```

In the 'Script Output' pane at the bottom, the output is:

```
BEGIN ex7('Horror'); END;
Error report -
ORA-20000: Nu exista filme cu genul specificat in baza de date. ← Yellow arrow pointing to this line
```

The 'Compiler - Log' tab in the 'Script Output' pane shows the error details:

```
ORA-06512: at "SYSTEM.EX7", line 71
ORA-06512: at line 1
20000. 00000 - "%s"
*Cause: The stored procedure 'raise_application_error'
        was called which causes this error to be generated.
```

8. Funcție care utilizează tabel auxiliar pentru gestionarea erorilor

Să se determine numărul total de filme difuzate de cinematografele specificate de utilizator, având o singură locație. Dacă compania cinematografului nu există în baza de date sau are mai multe locații, se vor trata situațiile corespunzătoare. Dintre aceste filme, să se calculeze procentul celor care au fost lansate după un anumit an.

-- Tabele utilizate: filme, cinematograf, difuzeaza

-- se creeaza un tabel pentru a tine evidenta erorilor aparute

`CREATE TABLE istoric_eroari(`

```
    id_eroare number generated by default as identity primary key,
    exercitiu varchar2(100),
    mesaj_eroare varchar2(4000),
    data_eroare varchar2(100)
);
```

The screenshot shows the Oracle SQL Developer interface. The left sidebar displays the project structure under 'PROJECT' with various database objects like Tables, Views, Procedures, etc. The central workspace shows a SQL Worksheet tab titled 'Ex 8.sql'. The code in the worksheet is:

```
-- Tabele utilizate: filme, cinematograf, difuzeaza

CREATE TABLE istoric_eroari(
    id_eroare number generated by default as identity primary key,
    exercitiu varchar2(100),
    mesaj_eroare varchar2(4000),
    data_eroare varchar2(100)
);
```

Below the code, the 'Script Output' window shows the message: "Table ISTORIC_ERORI created." with a yellow arrow pointing to it. The bottom status bar indicates "Line 19 Column 3".

-- se va utiliza o procedura pentru a putea insera mesajul erorilor in tabel

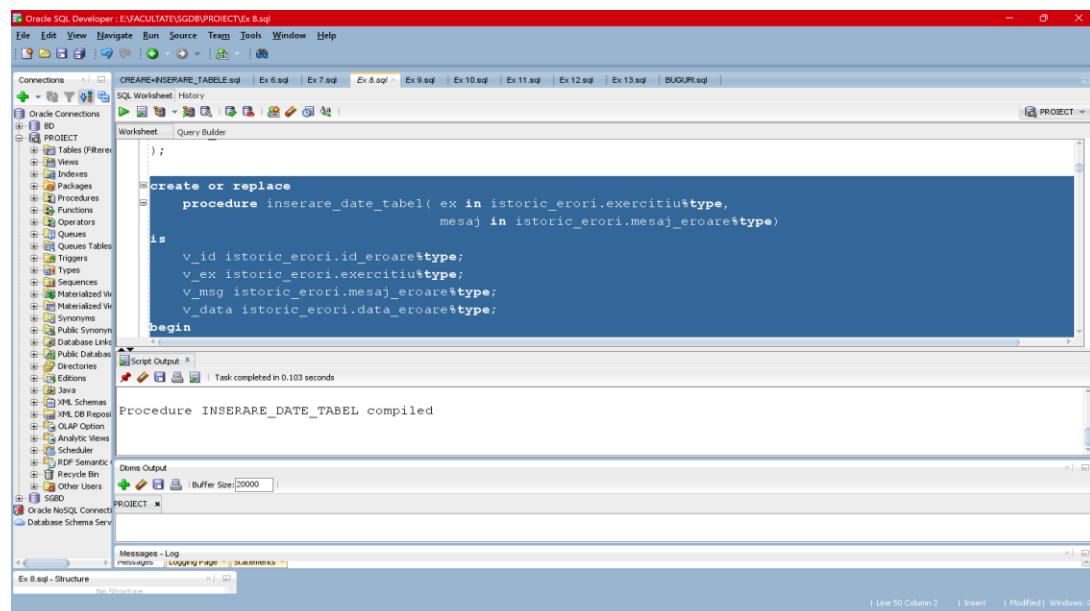
```

-- procedura se va apela in functia ex8 declarata mai jos
create or replace
procedure inserare_date_tabel( ex in istoric_erori.exercitiu%type,
                               mesaj in istoric_erori.mesaj_eroare%type) is
v_id istoric_erori.id_eroare%type;
v_ex istoric_erori.exercitiu%type;
v_msg istoric_erori.mesaj_eroare%type;
v_data istoric_erori.data_eroare%type;
begin
  INSERT INTO istoric_erori (exercitiu, mesaj_eroare, data_eroare)
    VALUES (ex, mesaj, to_char(sysdate, 'dd-mon-yyyy hh24:mi'))
  RETURNING id_eroare into v_id;

  -- preluam ultima linie bazata pe id-ul inserat
  select exercitiu, mesaj_eroare, data_eroare
  into v_ex, v_msg, v_data
  from istoric_erori
  where id_eroare = v_id;

  -- afisam eroarea inserata
  dbms_output.put_line('-----');
  dbms_output.put_line('Ultima eroare inserata');
  dbms_output.put_line('-----');
  dbms_output.put_line('Exercitiu: ' || v_ex);
  dbms_output.put_line('Mesaj eroare: ' || v_msg);
  dbms_output.put_line('Data eroare: ' || v_data);
end inserare_date_tabel;
/

```



create or replace function

```
ex8( nume in cinematograf.nume_companie%type,
      an in film.an_aparitie%type,
      total_filme out number)
      return number --se va returna procentul din enunt
is
  v_cod cinematograf.cod_cinematograf%type;
  mesaj_specific varchar2(4000);
  total_filme_dupa_anul_dat number := 0;
  NU_EXISTA_FILME EXCEPTION;
  AN_INVALID1 EXCEPTION;
  AN_INVALID2 EXCEPTION;
begin
  --tratarea exceptiilor NO_DATA_FOUND si TOO_MANY_ROWS cu ajutorul variabilei v_cod ←
  select cod_cinematograf into v_cod
    from cinematograf
   where nume_companie = nume;

  if an > extract(year from sysdate) then
    RAISE AN_INVALID1;
  end if;

  if an<1000 or an>9999 then
    RAISE AN_INVALID2;
  end if;

  -- UTILIZAREA A 3 TABELE INTR-O SINGURA COMANDA SQL ←
  -- determinare numar total filme
  select count(d.cod_film) into total_filme
    from film f
   join difuzeaza d on f.cod_film = d.cod_film
   join cinematograf c on d.cod_cinematograf = c.cod_cinematograf
  where c.cod_cinematograf = v_cod;

  -- determinarea numarului de filme care au aparut dupa un anumit an
  select count(f.cod_film) into total_filme_dupa_anul_dat
    from film f
   join difuzeaza d on f.cod_film = d.cod_film
   join cinematograf c on d.cod_cinematograf = c.cod_cinematograf
  where c.cod_cinematograf = v_cod and f.an_aparitie >= an;

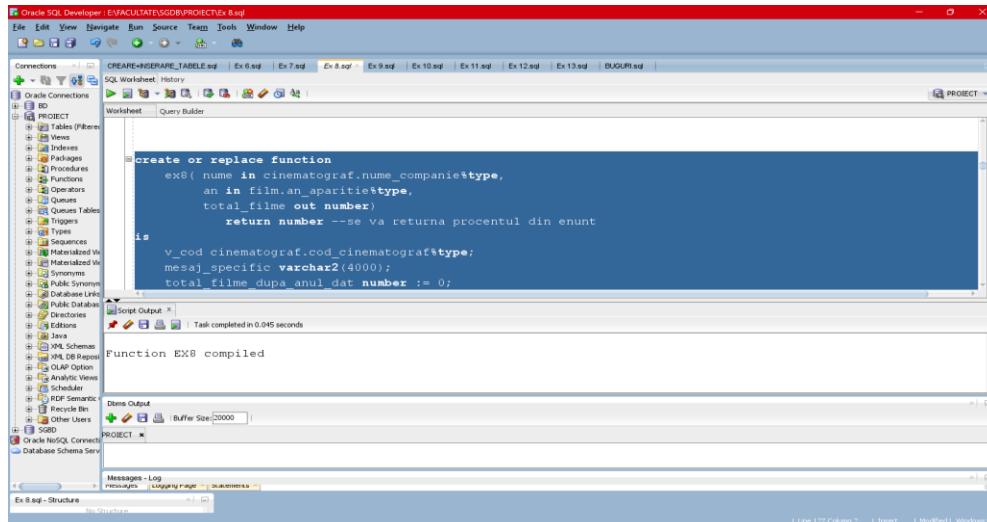
  if total_filme_dupa_anul_dat = 0 then
    RAISE NU_EXISTA_FILME;
```

```

end if;
return round(total_filme_dupa_anul_dat*100/total_filme,2);

EXCEPTION
WHEN NO_DATA_FOUND THEN
mesaj_specific := 'Nu s-a gasit in baza de date o companie cu numele ' || nume;
inserare_date_tabel('ex8', mesaj_specific);
return -1;
WHEN TOO_MANY_ROWS THEN
mesaj_specific := 'Exista in baza de date mai multe companii cu numele ' || nume;
inserare_date_tabel('ex8', mesaj_specific);
return -1;
WHEN NU_EXISTA_FILME THEN
mesaj_specific := 'Nu exista in baza de date filme care au aparut dupa anul ' || an;
inserare_date_tabel('ex8', mesaj_specific);
return 0;
WHEN AN_INVALID1 THEN
mesaj_specific := 'Anul introdus nu trebuie sa fie mai mare decat anul curent.';
inserare_date_tabel('ex8', mesaj_specific);
return -1;
WHEN AN_INVALID2 THEN
mesaj_specific := 'Anul introdus trebuie sa aiba 4 cifre.';
inserare_date_tabel('ex8', mesaj_specific);
return -1;
WHEN OTHERS THEN
mesaj_specific := 'A aparut alta eroare: ' || SQLERRM;
inserare_date_tabel('ex8', mesaj_specific);
return -1;
end ex8;
/

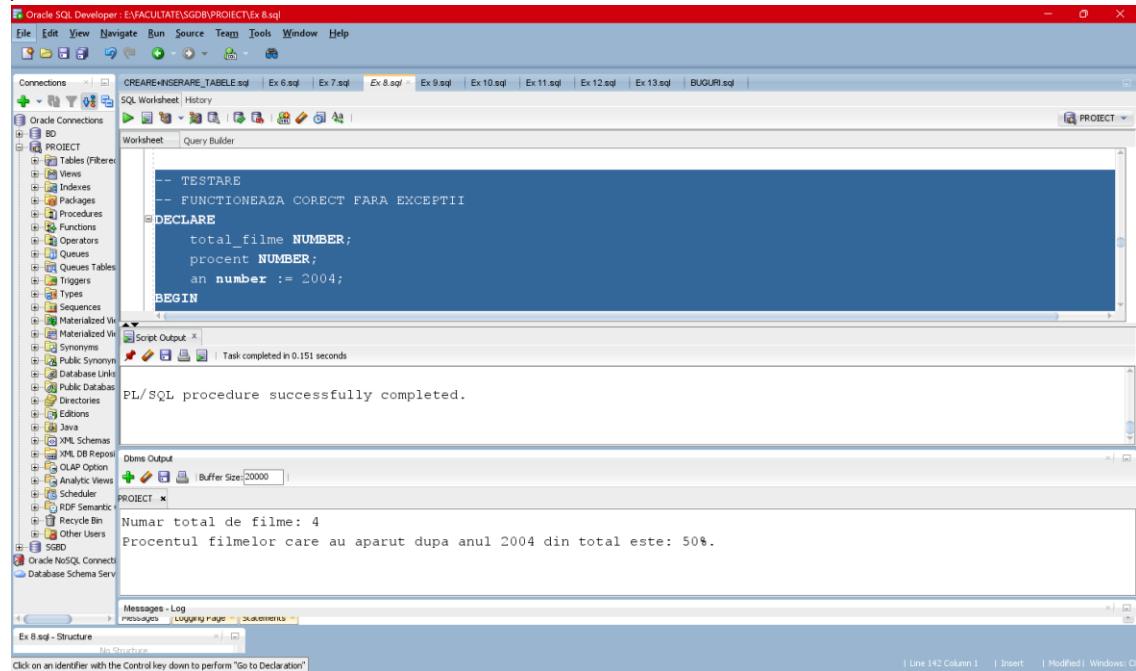
```



Testare functie ex8

```
-- FUNCTIONEAZA CORECT FARA EXCEPTII
-- prima varianta
DECLARE
    total_filme NUMBER;
    procent NUMBER;
    an number := 2004;
BEGIN
    procent := ex8('BV Cinemas', an, total_filme);

    DBMS_OUTPUT.PUT_LINE('Numar total de filme: ' || total_filme);
    DBMS_OUTPUT.PUT_LINE('Procentul filmelor care au aparut dupa anul ' || an || ' din total este:
    ' || procent || '%.');
END;
/
```



-- a doua varianta

```
VARIABLE procent NUMBER
VARIABLE total_filme NUMBER
EXECUTE :procent := ex8('GJ Cinemas', 2004, :total_filme)
PRINT procent
PRINT total_filme
```

```

-- a doua varianta
VARIABLE procent NUMBER
VARIABLE total_filme NUMBER
/

```

PL/SQL procedure successfully completed.

```

PROCENT
-----
      33.33

TOTAL_FILME
-----
          3

```

```

-- NO_DATA_FOUND
DECLARE
    total_filme NUMBER;
    procent NUMBER;
    an number := 2004;
BEGIN
    procent := ex8('Cinemas', an, total_filme);
END;
/

```

```

-- NO_DATA_FOUND
DECLARE
    total_filme NUMBER;
    procent NUMBER;

```

PL/SQL procedure successfully completed.

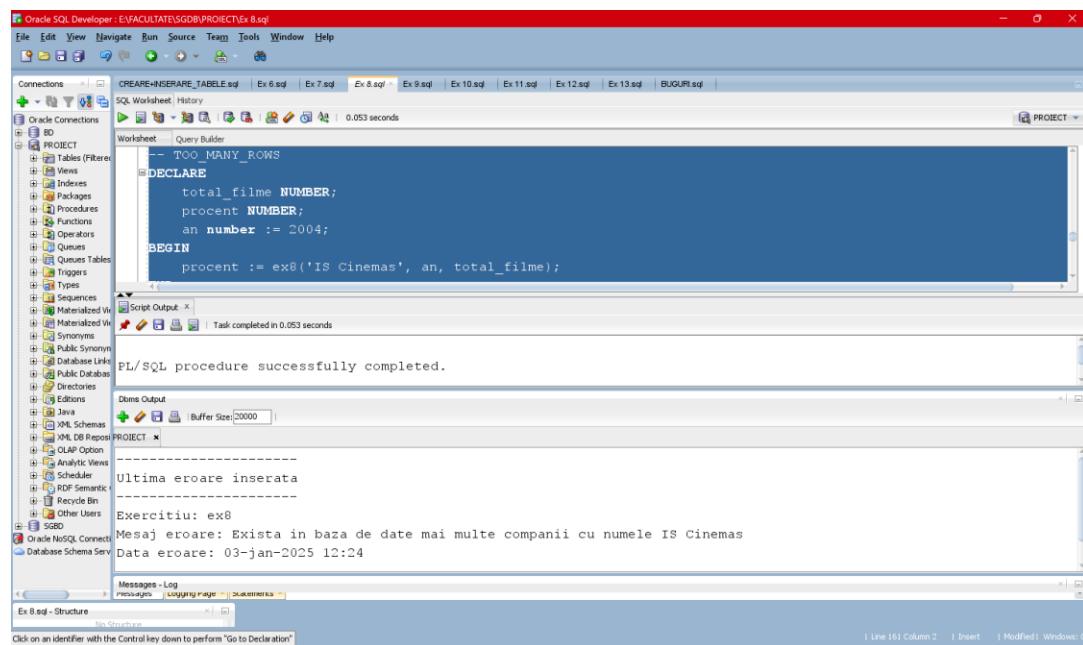
```

PROJECT *
-----
Ultima eroare inserata
-----
Exercitiu: ex8
Mesaj eroare: Nu s-a gasit in baza de date o companie cu numele Cinemas
Data eroare: 03-jan-2025 12:18

```

```
-- TOO_MANY_ROWS
```

```
DECLARE
    total_filme NUMBER;
    procent NUMBER;
    an number := 2004;
BEGIN
    procent := ex8('IS Cinemas', an, total_filme);
END;
/
```



```
-- NU_EXISTA_FILME
```

```
DECLARE
    total_filme NUMBER;
    procent NUMBER;
    an number := 2020;
BEGIN
    procent := ex8('BV Cinemas', an, total_filme);
    DBMS_OUTPUT.PUT_LINE('Numar total de filme: ' || total_filme);
    DBMS_OUTPUT.PUT_LINE('Procentul filmelor care au aparut dupa anul ' || an || ' din total este:
    ' || procent || '%.');
END;
/
```

```

-- AN_INVALID1
DECLARE
    total_filme NUMBER;
    procent NUMBER;
    an number := 2020;
BEGIN
    procent := ex8('BV Cinemas', an, total_filme);
END;
/

```

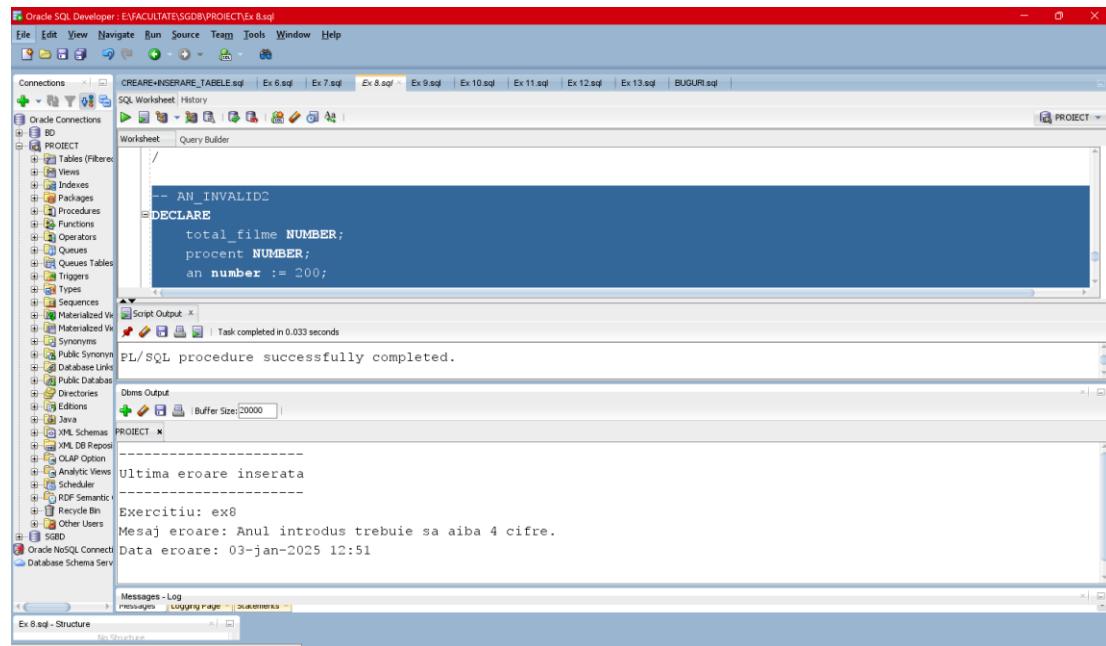
```

-- AN_INVALID1
DECLARE
    total_filme NUMBER;
    procent NUMBER;
    an number := 2026;
BEGIN
    procent := ex8('BV Cinemas', an, total_filme);
END;
/

```

```
-- AN_INVALID2

DECLARE
    total_filme NUMBER;
    procent NUMBER;
    an number := 200;
BEGIN
    procent := ex8('BV Cinemas', an, total_filme);
END;
/
```



```
-- VALUE_ERROR

BEGIN
    DECLARE
        total_filme NUMBER;
        procent NUMBER;
        an number := 'doua mii douazeci';
    BEGIN
        procent := ex8('BV Cinemas', an, total_filme);
    END;
EXCEPTION
    WHEN VALUE_ERROR THEN
        inserare_date_tabel('ex8', 'Anul introdus nu este valid, acesta trebuie sa fie de tipul numeric.');
    END;
/
```

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 8.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections Oracle Connections BD PROJECT Tables (Filtered) Views Indexes Packages Procedures Functions Operators Queues Queues Tables Triggers Types Sequences Materialized Views Materialized Views Synonyms Public Synonyms Database Links Public Databases Directories Editions Java XML Schemas XML DB Repos OLAP Option Analytic Views Scheduler RDF Semantic Recycle Bin Other Users SGBD Oracle NoSQL Connect Database Schema Serv

SQL Worksheet History

Worksheet Query Builder

```

END;
/
-- VALUE_ERROR
BEGIN
    DECLARE
        total_filme NUMBER;
        procent NUMBER;

```

Script Output x Task completed in 0.024 seconds

PL/SQL procedure successfully completed.

Doms Output

PROJECT x

Ultima eroare inserata

Exercitiu: ex8
Mesaj eroare: Anul introdus nu este valid, acesta trebuie sa fie de tipul numeric.
Data eroare: 03-jan-2025 12:53

Messages - Log

```
select *
from istoric_eroari;
```

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 8.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections Oracle Connections BD PROJECT Tables (Filtered) Views Indexes Packages Procedures Functions Operators Queues Queues Tables Triggers Types Sequences Materialized Views Materialized Views Synonyms Public Synonyms Database Links Public Databases Directories Editions Java XML Schemas XML DB Repos OLAP Option Analytic Views Scheduler RDF Semantic Recycle Bin Other Users SGBD Oracle NoSQL Connect Database Schema Serv

SQL Worksheet History

Worksheet Query Builder

```

select *
from istoric_eroari;

```

Script Output x Query Result x All Rows Fetched: 13 in 0.034 seconds

ID_EROARE	EXERCITIU	MESAJ_EROARE	DATA_EROARE
1	1ex8	Nu s-a gasit in baza de date o companie cu numele Cinemas	03-jan-2025 12:18
2	2ex8	Nu s-a gasit in baza de date o companie cu numele Cinemas	03-jan-2025 12:18
3	3ex8	Există in baza de date mai multe companii cu numele IS Cinemas	03-jan-2025 12:24
4	4ex8	Nu există in baza de date filme care au aparut dupa anul 2020	03-jan-2025 12:25
5	5ex8	Nu există in baza de date filme care au aparut dupa anul 2020	03-jan-2025 12:28
6	6ex8	Nu există in baza de date filme care au aparut dupa anul 2020	03-jan-2025 12:28
7	7ex8	Nu există in baza de date filme care au aparut dupa anul 2025	03-jan-2025 12:30
8	8ex8	Anul introdus nu trebuie sa fie mai mare decat anul curent.	03-jan-2025 12:36
9	9ex8	Anul introdus trebuie sa aiba 4 cifre.	03-jan-2025 12:51
10	10ex8	Anul introdus trebuie sa aiba 4 cifre.	03-jan-2025 12:51
11	11ex8	Anul introdus nu este valid, acesta trebuie sa fie de tipul numeric.	03-jan-2025 12:53
12	12ex8	Există in baza de date mai multe companii cu numele IS Cinemas	03-jan-2025 12:53
13	13ex8	Nu s-a gasit in baza de date o companie cu numele B Cinemas	03-jan-2025 14:24

Doms Output

PROJECT x

Messages - Log

9. Procedură care utilizează 4 parametrii și 3 exceptii proprii

Să se determine următoarele informații pentru filmele specificate de utilizatori, care sunt difuzate într-o singură zi în cinematograful indicat:

1. *Numărul de rezervări făcute pentru film.*
2. *Venitul total generat de rezervările pentru film.*
3. *Pentru clienții care au realizat rezervări:*
 - a. *Să se calculeze prețul rezervării, ținând cont de punctele bonus acumulate.*
4. *Să se clasifice acești clienți în funcție de punctele bonus, după cum urmează:*
 - a. *Client ocazional: între 100 și 150 puncte bonus.*
 - b. *Client activ: între 150 și 190 puncte bonus.*
 - c. *Client loial: peste 190 puncte bonus.*

-- Tabele utilizate: cinematograf, difuzeaza, film, rezervare, clienti

```
create or replace procedure
ex9(id_film in film.cod_film%type,
     id_cinema in cinematograf.cod_cinematograf%type,
     nr_rezervari out number, --determinarea punctului 1 din cerinta
     categorie out varchar2) -- determinarea punctului 4 din cerinta
is
    venit_total number; --variabila pentru determinarea punctului 2 din cerinta
    mesaj_istoric_eroari.mesaj_eroare%type;
    exista_film film.nume_film%type;
    exista_cinema number;
    exista_rezervari number;
    exista_o_zi date;
    eroare_client varchar2(40);
```

```
--cursor pentru a determina orele la care se difuzeaza filmul avand codul id_film
cursor c_ore is
    select t.ora_inceput inceput, t.ora_final final
    from difuzeaza d, table(d.perioade_difuzare) t
    where d.cod_film = id_film;
```

```
--cursor pentru a putea afisa detalii despre rezervarile clientilor (punctul 3 din cerinta)
cursor c_rezervari is
    select
        cl.nume_client as nume,
        cl.prenume_client as prenume,
```

```

cl.puncte_bonus as puncte,
r.pret_bilet*r.numar_persoane as pret_initial,
r.pret_bilet*r.numar_persoane - cl.puncte_bonus/10 as pret_final
from cinematograf c
join difuzeaza d on c.cod_cinematograf = d.cod_cinematograf
join film f on d.cod_film = f.cod_film
join rezervare r on f.cod_film = r.cod_film
join clienti cl on r.cod_client = cl.cod_client
where f.cod_film = id_film and c.cod_cinematograf = id_cinema;

```

-- Declarare exceptii proprii

```

NU_EXISTA_CINEMA EXCEPTION;
NU_EXISTA_REZERVARI EXCEPTION;
FARA_CATEGORIE EXCEPTION;

```

begin

```

-- tratarea erorii ca nu exista film cu acel cod --> NO_DATA_FOUND
select nume_film into exista_film
from film
where cod_film = id_film
and rownum = 1;

```

```

-- tratarea erorii ca nu exista cinematograf cu acel cod
select count(*) into exista_cinema
from cinematograf
where cod_cinematograf = id_cinema;

```

```

if exista_cinema = 0 then
    RAISE NU_EXISTA_CINEMA;
end if;

```

-- tratarea erorii ca filmul avand codul id_film nu se difuzeaza intr-o singura zi -->

TOO_MANY_ROWS

```

select distinct t.data_difuzare into exista_o_zi
from difuzeaza d, table(d.perioade_difuzare) t
where d.cod_film = id_film
and d.cod_cinematograf = id_cinema;

```

--tratarea erorii ca filmul se difuzeaza intr-o singura zi, dar nu are rezervari atribuite

```

select count(*) into nr_rezervari
from difuzeaza d

```

```

join film f on d.cod_film = f.cod_film
join rezervare r on f.cod_film = r.cod_film
where d.cod_film = id_film and d.cod_cinematograf = id_cinema;

if nr_rezervari = 0 then
    RAISE NU_EXISTA_REZERVARI;
end if;

dbms_output.put_line("");
dbms_output.put_line("      - DETALII DESPRE FILM -      ");
dbms_output.put_line("");
dbms_output.put_line('Filmul '||exista_film||' se difuzeaza in ziua '||exista_o_zi||' si in
intervalul orar: ');
for i in c_ore loop
    dbms_output.put_line(' - '||to_char(i.inceput, 'HH24:MI')||' - '||to_char(i.FINAL,
'HH24:MI'));
end loop;
dbms_output.new_line;

dbms_output.put_line("");
dbms_output.put_line("      - DETALII REZERVARI -      ");
dbms_output.put_line("");
dbms_output.put_line('Pentru acest film s-au facut '||nr_rezervari||' rezervari.');

dbms_output.put_line(' In functie de punctele bonus acumulate, clientii au obtinut
urmatoarele reduceri: ');
dbms_output.new_line;
for i in c_rezervari loop
    dbms_output.put_line('Numele clientului: '||i.nume||' '||i.prenume);
    dbms_output.put_line('Puncte bonus: '||i.puncte);
    dbms_output.put_line('Reducere: '||i.puncte/10||' lei');
    dbms_output.put_line('Pretul initial al rezervarii: '||i.pret_initial||' lei');
    dbms_output.put_line('Pretul final al rezervarii: '||i.pret_final||' lei');

    if
        i.puncte >= 100 and i.puncte < 150 then
            categorie := 'Ocazional';
    elsif
        i.puncte >= 150 and i.puncte < 190 then
            categorie := 'Activ';
    elsif
        i.puncte >= 190 then
            categorie := 'Loial';
    else

```

```

eroare_client := i.nume || ' ' || i.prenume;
RAISE FARA_CATEGORIE;
end if;
dbms_output.put_line('Categoria clientului: ' || categorie);
dbms_output.put_line('-----');
dbms_output.new_line;
end loop;

-- determinarea venitului total atribuit filmului avand codul id_film
-- UTILIZARE A 5 TABELE INTR-O SINGURA COMANDA SQL 
select
    trunc(sum(r.pret_bilet*r.numar_persoane - cl.puncte_bonus/10)) into venit_total
from cinematograf c
    join difuzeaza d on c.cod_cinematograf = d.cod_cinematograf
    join film f on d.cod_film = f.cod_film
    join rezervare r on f.cod_film = r.cod_film
    join clienti cl on r.cod_client = cl.cod_client
where f.cod_film = id_film and c.cod_cinematograf = id_cinema;
dbms_output.put_line('Filmul ' || exists_film || ' a adus incasari in valoare de ' || venit_total || ' lei.');

```

EXCEPTION

```

WHEN NO_DATA_FOUND THEN
mesaj := 'Nu s-a gasit in baza de date un film avand codul ' || id_film;
inserare_date_tabel('ex9', mesaj);
WHEN TOO_MANY_ROWS THEN
mesaj := 'Filmul care are codul ' || id_film || ' se difuzeaza in mai multe zile';
inserare_date_tabel('ex9', mesaj);
WHEN NU_EXISTA_CINEMA THEN
mesaj := 'In baza de date nu exista un cinematograf care sa aiba codul: ' || id_cinema;
inserare_date_tabel('ex9', mesaj);
WHEN NU_EXISTA_REZERVARI THEN
mesaj := 'In baza de date nu exista rezervari care sa includa filmul avand codul: ' || id_film
        || ' cu toate ca se difuzeaza intr-o singura zi.';
inserare_date_tabel('ex9', mesaj);
WHEN FARA_CATEGORIE THEN
mesaj := 'Clientul ' || eroare_client || ' nu apartine niciunei categorii.!';
inserare_date_tabel('ex9', mesaj);
WHEN OTHERS THEN
mesaj := 'A aparut alta eroare: ' || SQLERRM;
inserare_date_tabel('ex9', mesaj);
end ex9;
/

```

The screenshot shows the Oracle SQL Developer interface. In the central workspace, a PL/SQL script is being written:

```

create or replace procedure
    ex9(id_film in film.cod_film%type,
        id_cinema in cinematograf.cod_cinematograf%type,
        nr_rezervari out number, --determinarea punctului 1 din cerinta
        categorie out varchar2) -- determinarea punctului 4 din cerinta
is
    venit_total number; --variabila pentru determinarea punctului 2 din cerinta
    mesaj_istoric_errori.mesaj_eroare%type;
    exista_film film.nume_film%type;
    exista_cinema number;
    exista_rezervari number;
    exista_o_zi date;
    eroare_client varchar2(40);

```

The script is saved as `Ex 9.sql`. Below the workspace, the status bar indicates the file was saved to `E:\FACULTATE\SGDB\PROJECT\Ex 9.sql`.

Testare procedura ex9

-- FUNCTIONEAZA FARA EXCEPTII

-- varianta 1

```

declare
    nr_rezervari number;
    categorie varchar2(20);
begin
    ex9('F9', 'C7', nr_rezervari, categorie);
end;
/

```

The screenshot shows the Oracle SQL Developer interface after executing the stored procedure. The output window displays the results:

```

PL/SQL procedure successfully completed.

- DETALII DESPRE FILM -
Filmul Avengers: Endgame se difuzeaza in ziua 04-DEC-24 si in intervalul orar:
- 15:00 - 17:30
- 18:00 - 20:30
- DETALII REZERVARI -
Pentru acest film s-au facut 1 rezervari.

```

The status bar at the bottom right shows the current line and column information: Line 176 Column 2 | Insert | Modified | Windows: 0.

```

begin
    ex9('F9', 'C7', :nr_rezervari, categorie);
end;

```

PL/SQL procedure successfully completed.

In functie de punctele bonus acumulate, clientii au obtinut urmatoarele reduceri:

Numele clientului: Diaconu Cristian
Puncte bonus: 198
Reducere: 19.8 lei
Pretul initial al rezervarii: 70 lei
Pretul final al rezervarii: 50.2 lei
Categoria clientului: Loial

Filmul Avengers: Endgame a adus incasari in valoare de 50 lei.

-- varianta 2

variable nr_rezervari number
variable categorie varchar2
execute ex9('F15', 'C3', :nr_rezervari, :categorie)

```

variable categorie varchar2
execute ex9('F15', 'C3', :nr_rezervari, :categorie)

```

PL/SQL procedure successfully completed.

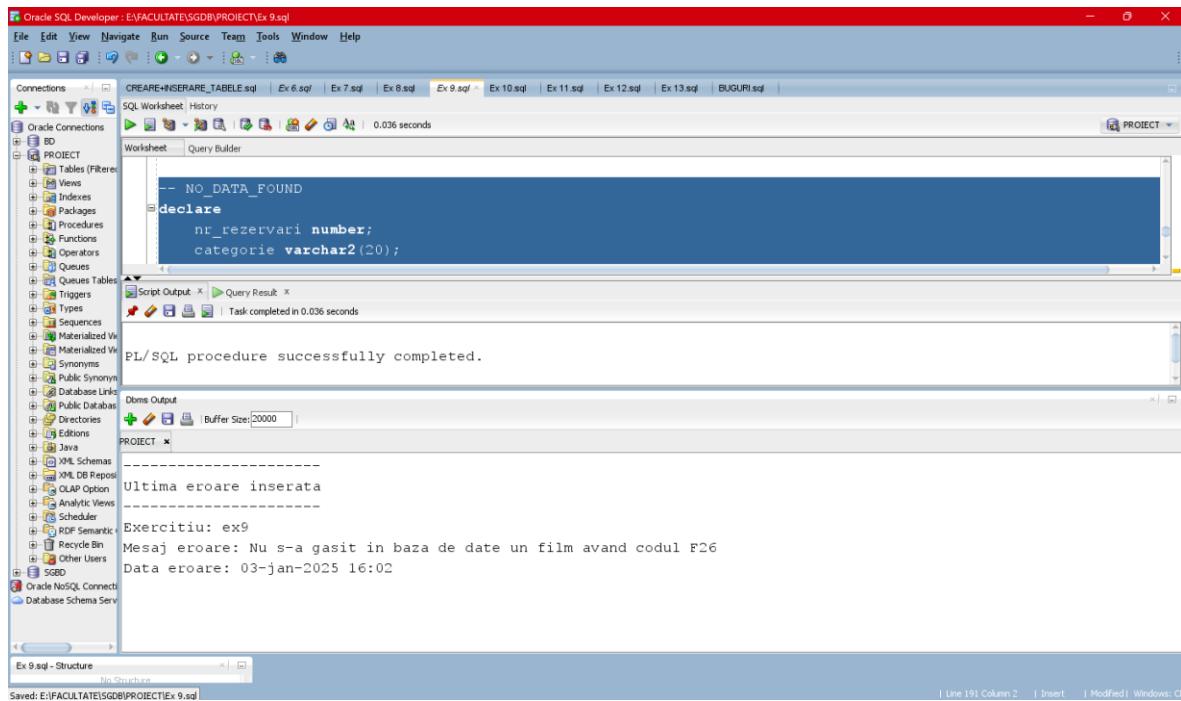
Pretul final al rezervarii: 17.5 lei
Categoria clientului: Activ

Numele clientului: Popescu Radu
Puncte bonus: 230
Reducere: 23 lei
Pretul initial al rezervarii: 100 lei
Pretul final al rezervarii: 77 lei
Categoria clientului: Loial

Filmul Forrest Gump a adus incasari in valoare de 176 lei.

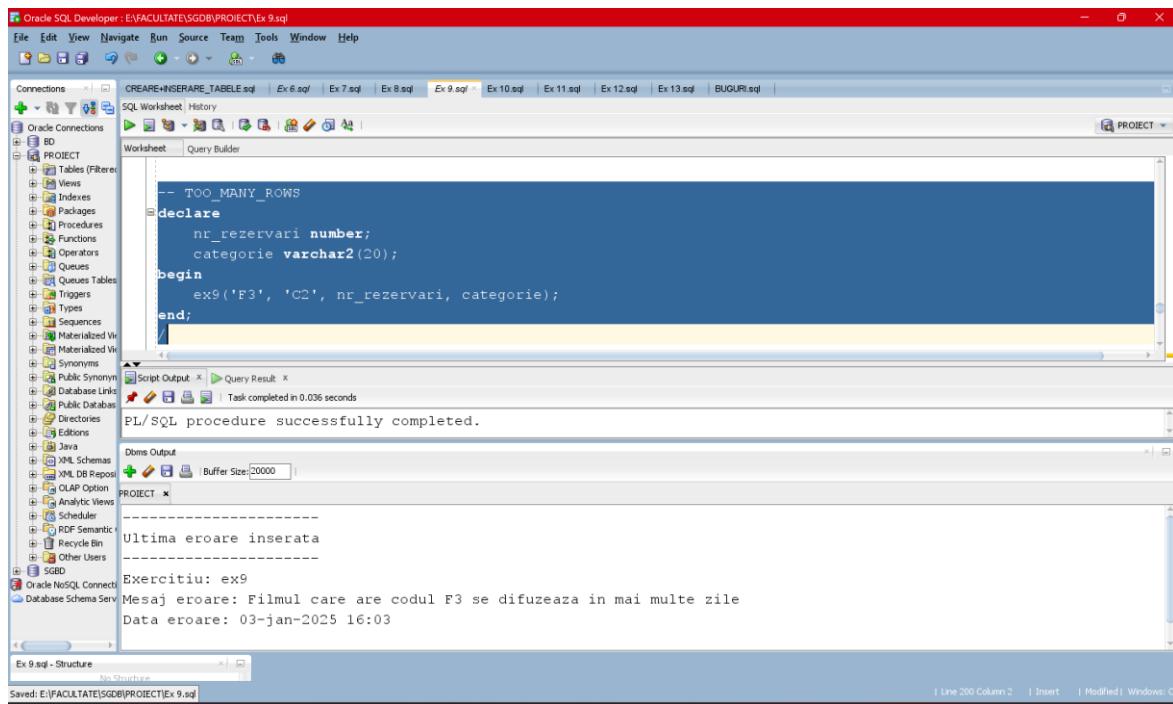
```
-- NO_DATA_FOUND
declare
    nr_rezervari number;
    categorie varchar2(20);
begin
    ex9('F26', 'C7', nr_rezervari, categorie);
end;
/

```



```
-- TOO_MANY_ROWS
declare
    nr_rezervari number;
    categorie varchar2(20);
begin
    ex9('F3', 'C2', nr_rezervari, categorie);
end;
/

```



```
-- VALUE_ERROR
begin
    declare
        nr_rezervari number := '';
        categorie varchar2(20);
    begin
        ex9('F3', 'C2', nr_rezervari, categorie);
    end;
EXCEPTION
    WHEN VALUE_ERROR THEN
        inserare_date_tabel('ex9', 'Al treilea argument al procedurii trebuie sa fie de tip
numeric.');
end;
/
```

```
-- VALUE_ERROR
begin
    declare
        nr_rezervari number := 1;
        categorie varchar2(20);
    begin
        ex9('F6', 'C11', nr_rezervari, categorie);
    end;
EXCEPTION
```

PL/SQL procedure successfully completed.

Exercitiu: ex9
Mesaj eroare: Al treilea argument al procedurii trebuie sa fie de tip numeric.
Data eroare: 03-jan-2025 16:05

```
-- NU_EXISTA_CINEMA
variable nr_rezervari number
variable categorie varchar2
execute ex9('F6', 'C11', :nr_rezervari, :categorie)
```

```
/
```

```
-- NU_EXISTA_CINEMA
variable nr_rezervari number
variable categorie varchar2
execute ex9('F6', 'C11', :nr_rezervari, :categorie)
```

PL/SQL procedure successfully completed.

Exercitiu: ex9
Mesaj eroare: In baza de date nu exista un cinematograf care sa aiba codul: C11
Data eroare: 03-jan-2025 16:08

```
-- NU_EXISTA_REZERVARI
variable nr_rezervari number
variable categorie varchar2
execute ex9('F25', 'C3', :nr_rezervari, :categorie)
```

```
-- NU_EXISTA_REZERVARI
variable nr_rezervari number
variable categorie varchar2
execute ex9('F25', 'C3', :nr_rezervari, :categorie)

-- FARA_CATEGORIE
```

PL/SQL procedure successfully completed.

Doms Output

Ultima eroare inserata

Exercitiu: ex9

Mesaj eroare: In baza de date nu exista rezervari care sa includa filmul avand codul: F25 cu toate ca se difuzeaza in

Data eroare: 03-jan-2025 16:08

```
-- FARA_CATEGORIE
variable nr_rezervari number
variable categorie varchar2
execute ex9('F23', 'C1', :nr_rezervari, :categorie)
```

```
variable categorie varchar2
execute ex9('F23', 'C1', :nr_rezervari, :categorie)
```

PL/SQL procedure successfully completed.

Doms Output

Numele clientului: Racovita Mihaela

Functie bonus: 10

Reducere: 1 lei

Pretul initial al rezervarii: 105 lei

Pretul final al rezervarii: 104 lei

Ultima eroare inserata

Exercitiu: ex9

Mesaj eroare: Clientul Racovita Mihaela nu apartine niciunei categorii. ←

Data eroare: 10-jan-2025 12:48

```
select * from istoric_erori;
```

--s-au adaugat in tabel exceptiile aparute in cadrul ex9

The screenshot shows the Oracle SQL Developer interface. In the top menu bar, the path is E:\FACULTATE\SGDB\PROJECT\Ex 9.sql. The main window has tabs for 'CREARE+INSERARE_TABLE.sq', 'Ex 6.sql', 'Ex 7.sql', 'Ex 8.sql', 'Ex 9.sql' (which is currently selected), 'Ex 10.sql', 'Ex 11.sql', 'Ex 12.sql', 'Ex 13.sql', and 'BUGURI.sql'. The 'Worksheet' tab is active, displaying the following PL/SQL code:

```
-- FARA_CATEGORIE
variable nr_rezervari number
variable categorie varchar2
execute ex9('F23', 'C1', :nr_rezervari, :categorie)

select * from istoric_erori;
```

Below the code, the 'Query Result' tab is open, showing the output of the query. A yellow arrow points to the first row of the result set, which is highlighted in blue. The result set contains 23 rows, each with columns ID_ERORE, EXERCITIU, and MESAJ_ERORE. The first row is:

ID_ERORE	EXERCITIU	MESAJ_ERORE
10	10ex8	Anul introdus trebuie sa aiba 4 cifre.
11	11ex8	Anul introdus nu este valid, acesta trebuie sa fie de tipul numeric.
12	12ex8	Există în baza de date mai multe companii cu numele IS Cinemas
13	13ex8	Nu s-a gasit în baza de date o companie cu numele B Cinemas
14	14ex9	A apărut alta eroare: ORA-06502: PL/SQL: numeric or value error: number precision too large
15	15ex9	Nu s-a gasit în baza de date un film având codul F26
16	16ex9	Filmul care are codul F3 se difuzează în mai multe zile
17	17ex9	Filmul care are codul F3 se difuzează în mai multe zile
18	18ex9	Al treilea argument al procedurii trebuie să fie de tip numeric.
19	19ex9	Al treilea argument al procedurii trebuie să fie de tip numeric.
20	20ex9	În baza de date nu există un cinematograf care să aibă codul: C11
21	21ex9	În baza de date nu există rezervări care să includă filmul având codul: F25 cu toate că se difuzează
22	22ex9	În baza de date nu există rezervări care să includă filmul având codul: F25 cu toate că se difuzează
23	23ex9	Cliențul Racovita Mihaela nu aparține niciunei categorii.

The status bar at the bottom right indicates 'Line 234 Column 29 | Insert | Modified | Windows: 0'.

10. Trigger de tip LMD la nivel de comandă

Se solicită crearea unui sistem de gestionare a angajărilor care să informeze utilizatorul despre toate operațiile efectuate asupra listei de angajați.

Sistemul trebuie să afișeze pe ecran detalii despre acțiunile efectuate, precum:

a. Adăugarea unui nou angajat (înregistrarea unei noi angajări).

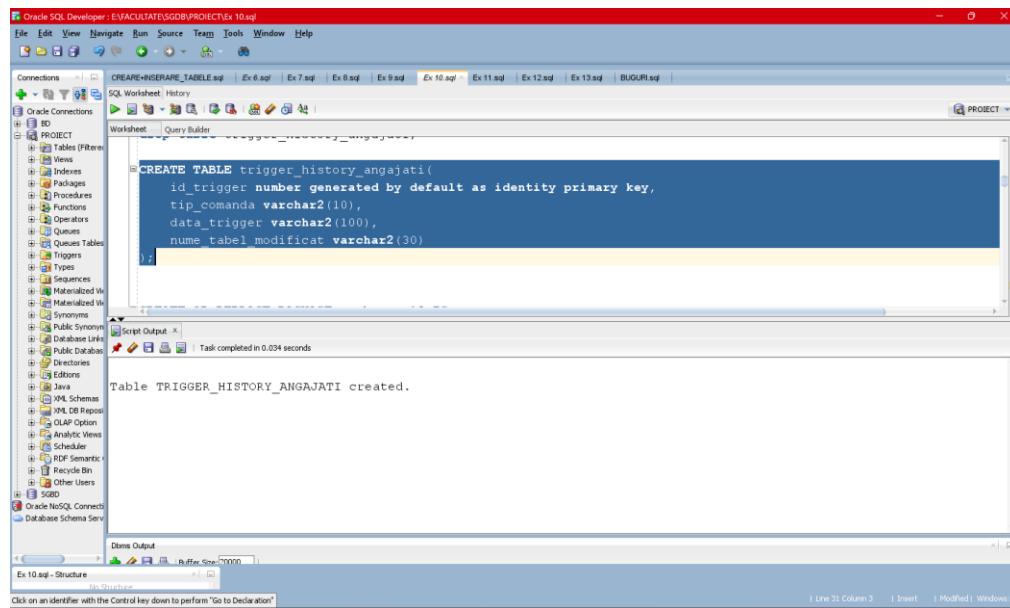
→ *Se va ține cont de faptul că data angajării nu poate fi în viitor.*

b. Eliminarea unui angajat din sistem (prin concediere sau demisie).

c. Modificarea informațiilor unui angajat existente în contractul de muncă (precum schimbări în funcție, salariu, departament etc.).

-- A fost implementat un pachet care contine procedurile necesare pentru validarea datei de angajare si inregistrarea detaliilor operatiunilor in tabelul trigger_history_angajati.
-- Verificarea datei de angajare a fost realizata prin intermediul unui trigger la nivel de linie,
-- iar ulterior a fost creat un trigger la nivel de comanda pentru a gestiona
-- inregistrarea operatiunilor efectuate.

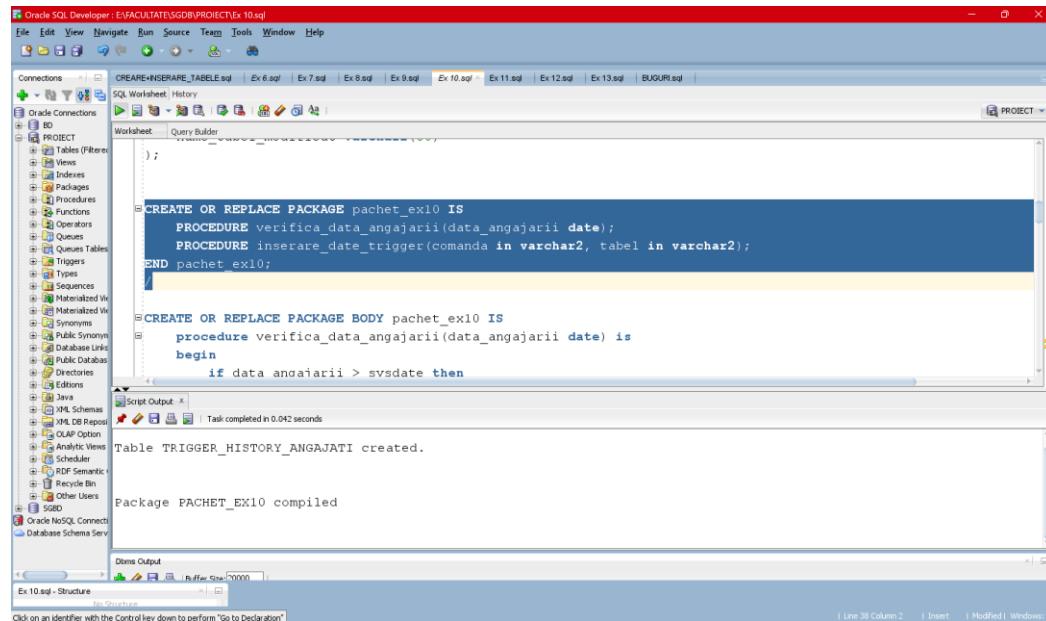
```
CREATE TABLE trigger_history_angajati(
    id_trigger number generated by default as identity primary key,
    tip_comanda varchar2(10),
    data_trigger varchar2(100),
    nume_tabel_modificat varchar2(30)
);
```



```

CREATE OR REPLACE PACKAGE pachet_ex10 IS
    PROCEDURE verifica_data_angajarii(data_angajarii date);
    PROCEDURE inserare_date_trigger(comanda in varchar2, tabel in varchar2);
END pachet_ex10;
/

```



```

CREATE OR REPLACE PACKAGE BODY pachet_ex10 IS
procedure verifica_data_angajarii(data_angajarii date) is
begin
if data_angajarii > sysdate then
    RAISE_APPLICATION_ERROR(-20010, 'Data angajarii nu poate fi in viitor!');
end if;
end verifica_data_angajarii;

procedure inserare_date_trigger(comanda in varchar2, tabel in varchar2)
is
v_id trigger_history_angajati.id_trigger%type;
v_comanda varchar2(10);
v_data varchar2(100);
v_tabel varchar2(30);
begin
insert into trigger_history_angajati(tip_comanda, data_trigger, nume_tabel_modificat)
values (comanda, to_char(sysdate, 'dd-mon-yyyy hh24:mi'), tabel)
returning id_trigger into v_id;

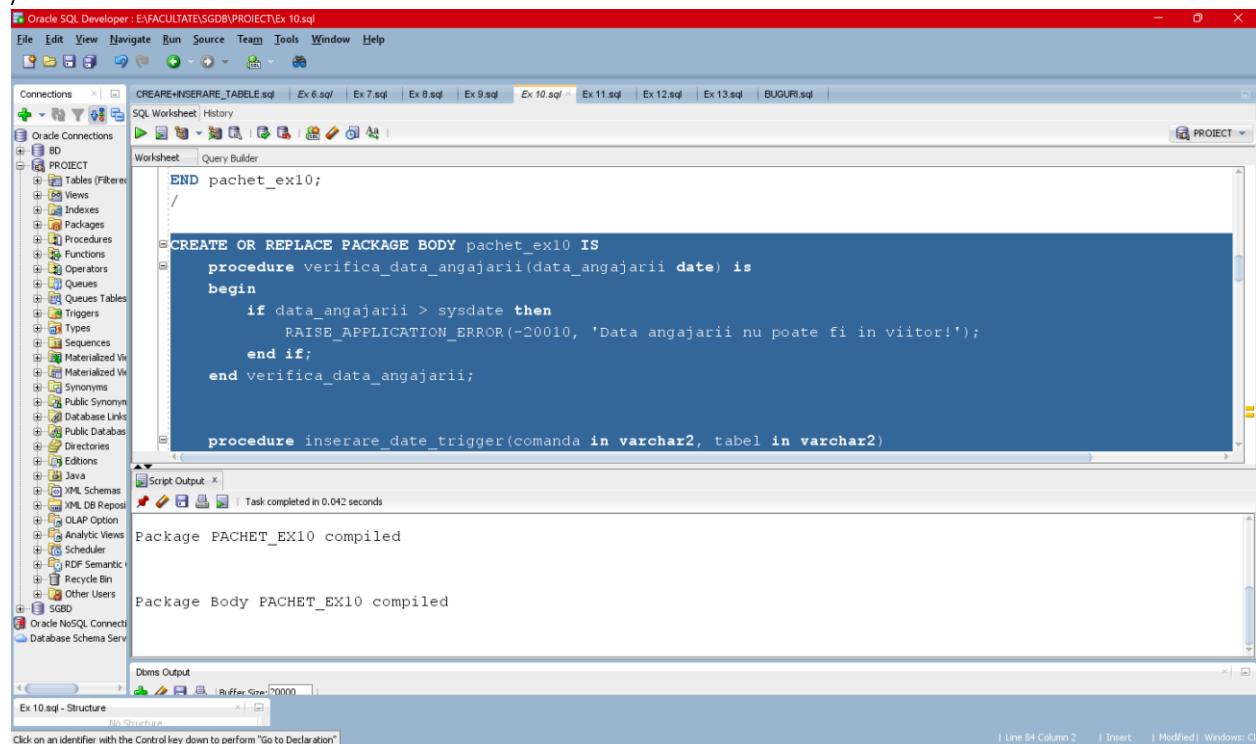
```

```

select data_trigger, nume_tabel_modificat
into v_data, v_tabel
from trigger_history_angajati
where id_trigger = v_id;

dbms_output.new_line;
dbms_output.put_line('-----');
dbms_output.put_line('Ultimul trigger declansat');
dbms_output.put_line('-----');
dbms_output.put_line('Tipul comenzii: ' || comanda);
dbms_output.put_line('Data declansarii triggerului: ' || v_data);
dbms_output.put_line('Numele tabelului asupra caruia s-au efectuat operatii: ' || v_tabel);
dbms_output.new_line;
end inserare_date_trigger;
END pachet_ex10;
/

```



```

CREATE OR REPLACE TRIGGER trigger_verif_data_angajarii
BEFORE INSERT on angajati
FOR EACH ROW
BEGIN
    pachet_ex10.verifica_data_angajarii(:NEW.data_angajare);
END;
/

```

```

CREATE OR REPLACE TRIGGER trigger_verif_data_angajarii
BEFORE INSERT on angajati
FOR EACH ROW
BEGIN
    pachet_ex10.verifica_data_angajarii(:NEW.data_angajare);
END;

```

Script Output: Task completed in 0.04 seconds

Package Body PACHET_EX10 compiled

Trigger TRIGGER_VERIF_DATA_ANGAJARII compiled

--Triggerul la nivel de comanda

```

CREATE OR REPLACE TRIGGER trigger_ex10
AFTER INSERT OR DELETE OR UPDATE ON angajati
BEGIN
    IF INSERTING THEN
        pachet_ex10.inserare_date_trigger('INSERT', 'Angajati');
    ELSIF DELETING THEN
        pachet_ex10.inserare_date_trigger('DELETE', 'Angajati');
    ELSIF UPDATING THEN
        pachet_ex10.inserare_date_trigger('UPDATE', 'Angajati');
    END IF;
END;
/

```

```

CREATE OR REPLACE TRIGGER trigger_ex10
AFTER INSERT OR DELETE OR UPDATE ON angajati
BEGIN
    IF INSERTING THEN
        pachet_ex10.inserare_date_trigger('INSERT', 'Angajati');
    ELSIF DELETING THEN
        pachet_ex10.inserare_date_trigger('DELETE', 'Angajati');
    ELSIF UPDATING THEN
        pachet_ex10.inserare_date_trigger('UPDATE', 'Angajati');
    END IF;
END;

```

Script Output: Task completed in 0.031 seconds

Trigger TRIGGER_VERIF_DATA_ANGAJARII compiled

Trigger TRIGGER_EX10 compiled

Testare trigger_ex10

-- verificare inserări

```
INSERT INTO angajati VALUES ('A28', 'J12', 'D7', 'C10','Florea', 'Alexandra', 2200, TO_DATE('20-DEC-24'), 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A29', 'J12', 'D7', 'C10','Florea', 'Alexandra', 2200, TO_DATE('05-JUN-25'), 'DD-MON-YY'),'Activ');
```

```
108
109
110
111 -- verificare inserari
112 INSERT INTO angajati VALUES ('A28', 'J12', 'D7', 'C10','Florea', 'Alexandra', 2200, TO_DATE('20-DEC-24'), 'DD-MON-YY'),'Activ');
113 INSERT INTO angajati VALUES ('A29', 'J12', 'D7', 'C10','Florea', 'Alexandra', 2200, TO_DATE('05-JUN-25'), 'DD-MON-YY'),'Activ');

1 row inserted.
```

```
from trigger_history_angajati;

-- verificare inserari
INSERT INTO angajati VALUES ('A28', 'J12', 'D7', 'C10','Florea', 'Alexandra', 2200, TO_DATE('20-DEC-24'), 'DD-MON-YY');
INSERT INTO angajati VALUES ('A29', 'J12', 'D7', 'C10','Florea', 'Alexandra', 2200, TO_DATE('05-JUN-25'), 'DD-MON-YY');

-- verificare update
update angajati
set cod_job = 'J11'
where cod_angajat = 'A28';

Error starting at line : 111 in command
INSERT INTO angajati VALUES ('A29', 'J12', 'D7', 'C10','Florea', 'Alexandra', 2200, TO_DATE('05-JUN-25'), 'DD-MON-YY'
Error report -
ORA-20010: Date angajarii nu poate fi in viitor!
ORA-06512: at "SYSTEM.PACHET_EX10", line 5
ORA-06512: at "SYSTEM.TRIGGER_VERIF_DATA_ANGAJARII", line 2
ORA-04088: error during execution of trigger 'SYSTEM.TRIGGER_VERIF_DATA_ANGAJARII'
```

-- drop trigger trigger_ex10;

```
select *
from angajati;
```

COD_ANGAJAT	COD_JOB	COD_DEPARTAMENT	COD_CINEMATOGRAF	NUME_ANGAJAT	PRENUME_ANGAJAT	SALARIU_ANGAJAT	DATA_ANGAJARE	STATUS_ANGAJAT
18 A18	J4	D4	C2	Popica	Tudor	4050	19-JUN-23	In concediu
19 A19	J5	D5	C10	Florea	Bianca	3300	15-JUL-20	Activ
20 A20	J11	D3	C1	Munteanu	Andrada	6400	27-AUG-22	Activ
21 A21	J8	D8	C2	Salciuanu	Stefan	2700	10-SEP-23	In concediu
22 A22	J12	D7	C3	Monceanu	Valentina	2600	15-AUG-22	Suspendat
23 A23	J8	D8	C4	Andruta	Andra	2900	27-JUL-20	Activ
24 A24	J9	D9	C9	Tanislav	Alexia	2650	08-JUN-21	In concediu
25 A25	J10	D6	C6	Velcea	Mihnea	3400	01-SEP-23	Activ
26 A26	J10	D6	C10	Lupu	Andrei	5000	12-OCT-23	Suspendat
27 A27	J11	D3	C8	Ciorita	Alexandra	6500	19-SEP-24	In concediu
28 A28	J12	D7	C10	Florea	Alexandra	2200	20-DEC-24	Activ

-- se vor mai insera 2 angajati, angajatul A29 si A30

```
INSERT INTO angajati VALUES ('A29', 'J12', 'D7', 'C10', 'XX', 'Alexandra', 2200, TO_DATE('05-JUN-24', 'DD-MON-YY'), 'Activ');
INSERT INTO angajati VALUES ('A30', 'J12', 'D7', 'C10', 'YY', 'Alexandra', 2200, TO_DATE('05-JUN-24', 'DD-MON-YY'), 'Activ');
```

```
102
103 select *
104 from angajati;
105
```

COD_ANGAJAT	COD_JOB	COD_DEPARTAMENT	COD_CINEMATOGRAF	NUME_ANGAJAT	PRENUME_ANGAJAT	SALARIU_ANGAJAT	DATA_ANGAJARE	STATUS_ANGAJAT
23 A23	J8	D8	C4	Andruta	Andra	2900	27-JUL-20	Activ
24 A24	J9	D9	C9	Tanislav	Alexia	2650	08-JUN-21	In concediu
25 A25	J10	D6	C6	Velcea	Mihnea	3400	01-SEP-23	Activ
26 A26	J10	D6	C10	Lupu	Andrei	5000	12-OCT-23	Suspendat
27 A27	J11	D3	C8	Ciorita	Alexandra	6500	19-SEP-24	In concediu
29 A29	J12	D7	C10	XX	Alexandra	2200	05-JUN-24	Activ
30 A30	J12	D7	YY	Alexandra	2200	05-JUN-24	Activ	

Ultimul trigger declansat

Tipul comenzii: INSERT

Data declansarii triggerului: 10-jan-2025 13:13

Numele tabelului asupra caruia s-a efectuat operatii: Angajati

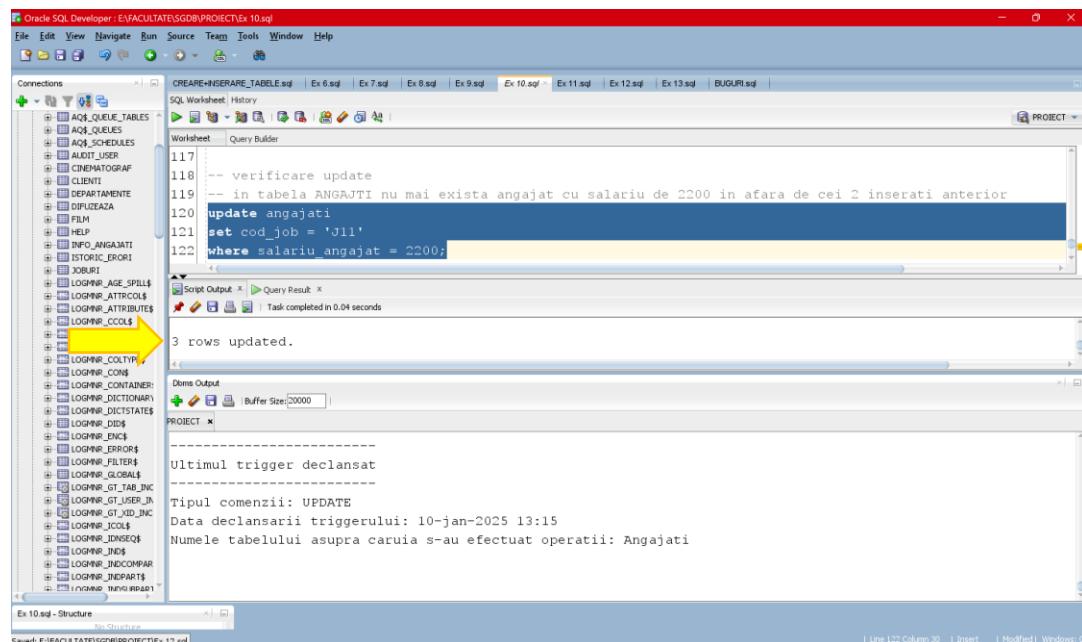
```
-- verificare update
```

```
-- angajatii A28, A29 si A30 sunt singurii care au salariul de 2200, prin urmare, urmatorul update  
-- va actualiza 3 linii in acelasi timp
```

```
update angajati
```

```
set cod_job = 'J11'
```

```
where salariu_angajat = 2200;
```



```
117  
118 -- verificare update  
119 -- in tabela ANGAJATI nu mai exista angajat cu salariu de 2200 in afara de cei 2 inserati anterior  
120 update angajati  
121 set cod_job = 'J11'  
122 where salariu_angajat = 2200;
```

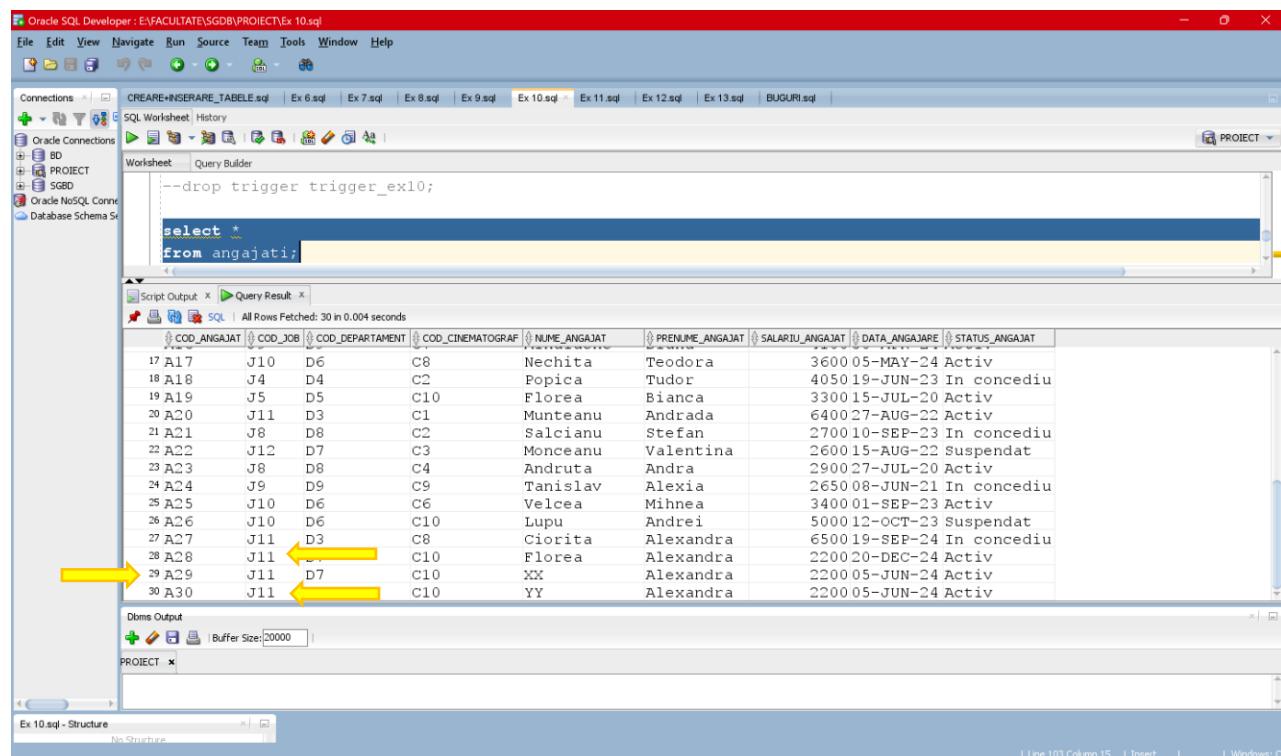
3 rows updated.

Ultimul trigger declansat

Tipul comenzi: UPDATE

Data declansarii triggerului: 10-jan-2025 13:15

Numele tabelului asupra caruia s-au efectuat operatii: Angajati



```
--drop trigger trigger_ex10;
```

```
select *  
from angajati;
```

COD_ANGAJAT	COD_JOB	COD_DEPARTAMENT	COD_CINEMATOGRAF	NUME_ANGAJAT	PRENUME_ANGAJAT	SALARIU_ANGAJAT	DATA_ANGAJARE	STATUS_ANGAJAT
17 A17	J10	D6	C8	Nechita	Teodora	3600	05-MAY-24	Activ
18 A18	J4	D4	C2	Popica	Tudor	4050	19-JUN-23	In concediu
19 A19	J5	D5	C10	Florea	Bianca	3300	15-JUL-22	Activ
20 A20	J11	D3	C1	Munteanu	Andrada	6400	27-AUG-22	Activ
21 A21	J8	D8	C2	Salcianu	Stefan	2700	10-SEP-23	In concediu
22 A22	J12	D7	C3	Monceanu	Valentina	2600	15-AUG-22	Suspendat
23 A23	J8	D8	C4	Andruta	Andra	2900	27-JUL-20	Activ
24 A24	J9	D9	C9	Tanislav	Alexia	2650	08-JUN-21	In concediu
25 A25	J10	D6	C6	Velcea	Mihnea	3400	01-SEP-23	Activ
26 A26	J10	D6	C10	Lupu	Andrei	5000	12-OCT-23	Suspendat
27 A27	J11	D3	C8	Ciorita	Alexandra	6500	19-SEP-24	In concediu
28 A28	J11	D3	C10	Florea	Alexandra	2200	20-DEC-24	Activ
29 A29	J11	D7	C10	XX	Alexandra	2200	05-JUN-24	Activ
30 A30	J11	D7	C10	YY	Alexandra	2200	05-JUN-24	Activ

-- in tabelul trigger_history_angajati vor aparea 3 inserturi si un singur update realizat

The screenshot shows the Oracle SQL Developer interface. In the top navigation bar, the file 'Ex 10.sql' is open. The left sidebar displays a tree view of database objects under 'Connections'. A yellow arrow points from the tree view to the 'Script Output' window, which contains the following SQL code:

```
102
103 select *
104   from angajati;
105
106 select *
107   from trigger_history_angajati;
108
109
110 -- verificare inserare
111
```

The 'Query Result' window shows a table with four rows of data:

ID_TRIGGER	TIP_COMANDA	DATA_TRIGGER	NUME_TABEL_MODIFICAT
1	INSERT	10-jan-2025 13:12	Angajati
2	INSERT	10-jan-2025 13:13	Angajati
3	INSERT	10-jan-2025 13:13	Angajati
4	UPDATE	10-jan-2025 13:15	Angajati

-- verificare delete

-- se vor sterge cei trei angajati, A28, A29, A30

delete from angajati

where salariu_angajat = 2200;

The screenshot shows the Oracle SQL Developer interface. In the top navigation bar, the file 'Ex 10.sql' is open. The left sidebar displays a tree view of database objects under 'Connections'. A yellow arrow points from the tree view to the 'Script Output' window, which contains the following SQL code:

```
121 set cod_job = 'J11'
122 where salariu_angajat = 2200;
123
124 -- verificare delete
125 delete from angajati
126 where salariu_angajat = 2200;
127
128
```

The 'Query Result' window shows the output of the delete command:

```
3 rows deleted.
```

The 'Doms Output' window shows the following messages:

```
-----  
Ultimul trigger declansat  
-----  
Tipul comenzii: DELETE  
Data declansarii triggerului: 10-jan-2025 13:16  
Numele tabelului asupra caruia s-au efectuat operatii: Angajati
```

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 10.sql

```

CREATE+INSERARE_TABLE.sql | Ex 6.sql | Ex 7.sql | Ex 8.sql | Ex 9.sql | Ex 10.sql | Ex 11.sql | Ex 12.sql | Ex 13.sql | BUGURI.sql |
File Edit View Navigate Run Source Team Tools Window Help
Connections BD PROJECT
  Oracle Connections
    + Tables (Filtered)
      + Views
      + Indexes
      + Packages
      + Procedures
      + Functions
      + Operators
      + Queues
      + Queues Tables
      + Triggers
      + Types
      + Sequences
      + Materialized Views
      + Materialized VIEWS
      + Synonyms
      + Public Synonyms
      + Database Links
      + Public Databases
      + Directories
      + Editions
      + Java
      + XML Schemas
      + XML DB Reposi
      + OLAP Optimi
      + Analytic Views
      + Scheduler
      + RDF Semantic
      + Recycle Bin
      + Other Users
        + Oracle 11g
        + Database Schema Services
          + Dims Output
            + Buffer Size: 20000
        + PROJECT
          + Ex 10.sql - Structure
            + No Structure
Click on an identifier with the Control key down to perform "Go to Declaration"

```

Worksheet Query Builder

```

pachet_exiu.inserare_date_trigger('UPDATE', 'Angajati');
END IF;
END;
/
--drop trigger trigger_ex10;

select *
from angajati;

```

Query Result

COD_ANGAJAT	COD_JOB	COD_DEPARTAMENT	COD_CINEMATOGRAF	NUME_ANGAJAT	PRENUME_ANGAJAT	SALARIU_ANGAJAT	DATA_ANGAJARE	STATUS_ANGAJAT
19 A19	J5	D5	C10	Florea	Bianca	3300	15-JUL-20	Activ
20 A20	J11	D3	C1	Munteanu	Andrada	6400	27-AUG-22	Activ
21 A21	J8	D8	C2	Salcianu	Stefan	2700	10-SEP-23	In concediu
22 A22	J12	D7	C3	Monceanu	Valentina	2600	15-AUG-22	Suspendat
23 A23	J8	D8	C4	Andruta	Andra	2900	27-JUL-20	Activ
24 A24	J9	D9	C9	Tanislav	Alexia	2650	08-JUN-21	In concediu
25 A25	J10	D6	C6	Velcea	Mihnea	3400	01-SEP-23	Activ
26 A26	J10	D6	C10	Lupu	Andrei	5000	12-OCT-23	Suspendat
27 A27	J11	D3	C8	Ciorita	Alexandra	6500	19-SEP-24	In concediu

--in tabelul trigger_history_angajati va aparea o singura operatie de delete, cu toate ca s-au efectuat 3 stergeri

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 10.sql

```

CREATE+INSERARE_TABLE.sql | Ex 6.sql | Ex 7.sql | Ex 8.sql | Ex 9.sql | Ex 10.sql | Ex 11.sql | Ex 12.sql | Ex 13.sql | BUGURI.sql |
File Edit View Navigate Run Source Team Tools Window Help
Connections BD PROJECT
  Oracle Connections
    + AQ$QUEUE_TABLES
      + AQ$QUEUES
      + AQ$SCHEDULES
      + AUDIT_USER
      + CINEMATOGRAF
      + CLIENTI
      + DEPARTAMENTE
      + DIFUZEAZA
      + FILM
      + HELP
      + INFO_ANGAJATI
      + ISTORIC_ERORI
      + JOBLIST
      + LOGMN$AGE_SPILL$
      + LOGMN$ATTRCOLS
      + LOGMN$ATTRIBUTES
      + LOGMN$CCOL$#
      + LOGMN$CFDEF#
      + LOGMN$COL$#
      + LOGMN$COLTYPE$#
      + LOGMN$CON$#
      + LOGMN$CONTAINER#
      + LOGMN$DICTIONARY#
      + LOGMN$DICTSTATE#
      + LOGMN$DIDE#
      + LOGMN$ENC#
      + LOGMN$ERRORS#
      + LOGMN$FILTER#
      + LOGMN$GLOBAL$#
      + LOGMN$GT_VAR_THC#
      + LOGMN$GT#
      + LOGMN$GT_XID_INC#
      + LOGMN$ICOL$#
      + LOGMN$INSEC$#
      + LOGMN$INDE#
      + LOGMN$INCOMPAT#
      + LOGMN$INDPARTS#
      + LOGMN$TRNG_IPRPT#
        + Oracle 11g
        + Database Schema Services
          + Dims Output
            + Buffer Size: 20000
        + PROJECT
          + Ex 10.sql - Structure
            + No Structure
Click on an identifier with the Control key down to perform "Go to Declaration"

```

Worksheet Query Builder

```

100
101 --drop trigger trigger_ex10;
102
103 select *
104   from angajati;
105
106 select *
107   from trigger history angajati;
108
109
110

```

Query Result

ID_TRIGGER	TIPO_COMANDA	DATA_TRIGGER	NUME_TABEL_MODIFICAT
1	1 INSERT	10-jan-2025 13:12	Angajati
2	2 INSERT	10-jan-2025 13:13	Angajati
3	3 INSERT	10-jan-2025 13:13	Angajati
4	4 UPDATE	10-jan-2025 13:15	Angajati
5	5 DELETE	10-jan-2025 13:16	Angajati

11. Trigger de tip LMD la nivel de linie (compound trigger)

Să se implementeze următoarele reguli pentru organizarea angajaților:

1. Suspendarea unui angajat

Când statutul angajatului este actualizat la "Suspendat", compania îi va reduce salariul cu 5%. Cu toate acestea, reducerea nu poate să ducă salariul angajatului sub nivelul minim prevăzut pentru jobul său. Dacă se ajunge la această situație, salariul angajatului va lua valoarea minimă menționată anterior.

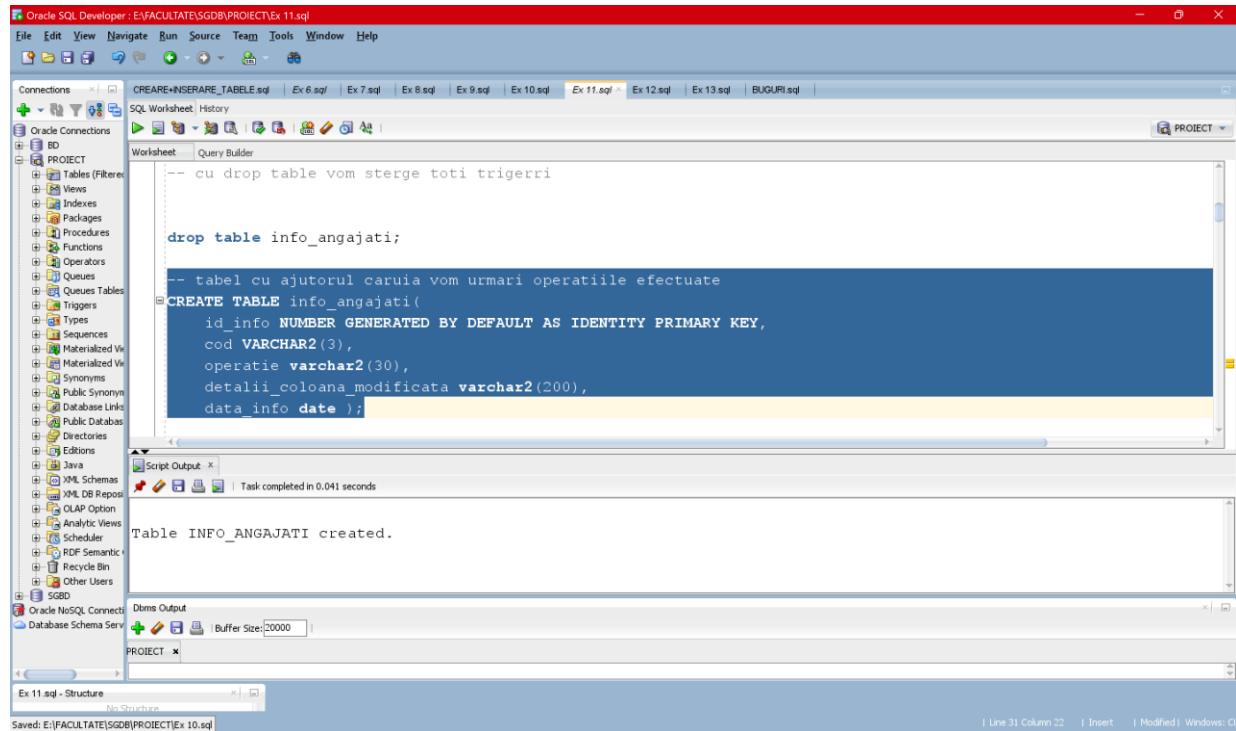
2. Adăugarea unui nou angajat

La momentul efectuării unei noi angajări se va ține cont de faptul că salariul trebuie să fie cel minim corespunzător jobului pentru care lucrează. De asemenea, dacă numărul maxim de angajați permis pentru cinematograf este depășit, se va afișa un mesaj de eroare care să informeze utilizatorul despre această situație.

-- tabel cu ajutorul caruia vom urmari operațiile efectuate

```
CREATE TABLE info_angajati(
```

```
    id_info NUMBER GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,
    cod VARCHAR2(3),
    operatie varchar2(30),
    detalii_coloana_modificata varchar2(200),
    data_info date );
```



The screenshot shows the Oracle SQL Developer interface with the following details:

- File Bar:** File, Edit, View, Navigate, Run, Source, Team, Tools, Window, Help.
- Connections:** BD, PROJECT.
- Tables:** Tables (Filter), Views, Indexes, Packages, Procedures, Functions, Operators, Queues, Queues Tables, Triggers, Types, Sequences, Materialized Views, Synonyms, Public Synonyms, Database Links, Public Databases, Directories, Editions, Java, XML Schemas, XML DB Repositories, OLAP Options, Analytic Views, Schedulers, RDF Semantic, Recycle Bin, Other Users, SGBD.
- Project:** PROJECT.
- Worksheet:** Shows the SQL code for creating the table:

```
-- cu drop table vom sterge toti trigerri
drop table info_angajati;

-- tabel cu ajutorul caruia vom urmari operațiile efectuate
CREATE TABLE info_angajati(
    id_info NUMBER GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,
    cod VARCHAR2(3),
    operatie varchar2(30),
    detalii_coloana_modificata varchar2(200),
    data_info date );
```
- Script Output:** Task completed in 0.041 seconds.
- Table Output:** Table INFO_ANAJATI created.
- DMS Output:** Buffer Size: 20000.
- Bottom Status Bar:** Line 31 Column 22 | Insert | Modified | Windows: C:\

```

CREATE OR REPLACE TRIGGER trigger_ex11

FOR INSERT OR UPDATE OF status_angajat ON angajati
COMPOUND TRIGGER
    record_info info_angajati%rowtype;
BEFORE EACH ROW IS
    minim joburi.salariu_minim%type;
    maxim_angajati constant number := 10;
    nr_angajati_in_prezent number;
BEGIN
    if UPDATING then
        if :NEW.status_angajat = 'Suspendat' then
            select salariu_minim into minim
            from joburi
            where cod_job = :OLD.cod_job;

            :NEW.salariu_angajat := :OLD.salariu_angajat - 0.05*:OLD.salariu_angajat;

            if :NEW.salariu_angajat < minim then
                :NEW.salariu_angajat := minim;
            end if;

            record_info.cod := :OLD.cod_angajat;
            record_info.operatie := 'Suspendare';
            record_info.detalii_coloana_modificata := 'Salariul a fost redus de la ' ||
                :OLD.salariu_angajat || ' la ' || :NEW.salariu_angajat
                || ', dar nu sub salariul minim (' || minim || ').';
            record_info.data_info := sysdate;
        end if;
    elsif INSERTING then
        select salariu_minim into minim
        from joburi
        where cod_job = :NEW.cod_job;

        if :NEW.salariu_angajat != minim then
            RAISE_APPLICATION_ERROR(-20001, 'Salariul trebuie sa fie cel minim corespunzator
jobului (' ||
                || minim || ').');
        end if;

        select count(*) into nr_angajati_in_prezent
        from angajati
        where cod_cinematograf = :NEW.cod_cinematograf;
    end if;
end;

```

```

if nr_angajati_in_prezent >= maxim_angajati THEN
    RAISE_APPLICATION_ERROR(-20002, 'Cinematograful a atins limita maxima de
angajati (' ||
    || maxim_angajati || '.');
end if;

record_info.cod := :NEW.cod_angajat;
record_info.operatie := 'Adaugare';
record_info.detalii_coloana_modificata := 'Angajat nou cu salariul minim corespunzator
jobului (' || minim || ')';
record_info.data_info := sysdate;
end if;
END BEFORE EACH ROW;

AFTER EACH ROW IS
BEGIN
    if record_info.cod is NOT NULL then
        insert into info_angajati(cod, operatie, detalii_coloana_modificata, data_info)
        values (record_info.cod, record_info.operatie,
record_info.detalii_coloana_modificata, record_info.data_info);

        dbms_output.put_line('Cod angajat: ' || record_info.cod);
        dbms_output.put_line('Operatie: ' || record_info.operatie);
        dbms_output.put_line('Detalii modificare: ' || record_info.detalii_coloana_modificata);
        dbms_output.put_line('Data operatiei: ' || to_char(record_info.data_info, 'DD-MON-
YYYY HH24:MI'));
    end if;

    -- resetam valorile
    record_info.cod := null;
    record_info.operatie := null;
    record_info.detalii_coloana_modificata := null;
    record_info.data_info := null;
END AFTER EACH ROW;
END;
/

```

```

CREATE OR REPLACE TRIGGER trigger_ex11
  FOR INSERT OR UPDATE OF status_angajat ON angajati
  COMPOUND TRIGGER
    record info info_angajati%rowtype;
    BEFORE EACH ROW IS
      data_info date;
    BEGIN
      data_info := sysdate;
      record.info.data := data_info;
    END;
  END;

```

Trigger TRIGGER_EX11 compiled

Testare trigger_ex11

-- verificare inserari corecte cu salariu corect

```

INSERT INTO angajati VALUES ('A28', 'J12', 'D7', 'C10', 'Florea', 'Alexandra', 2200, TO_DATE('20-DEC-24', 'DD-MON-YY'), 'Activ');
INSERT INTO angajati VALUES ('A29', 'J12', 'D7', 'C10', 'Onila', 'Matei', 2200, TO_DATE('21-DEC-24', 'DD-MON-YY'), 'Activ');

```

```

1 row inserted.

1 row inserted.

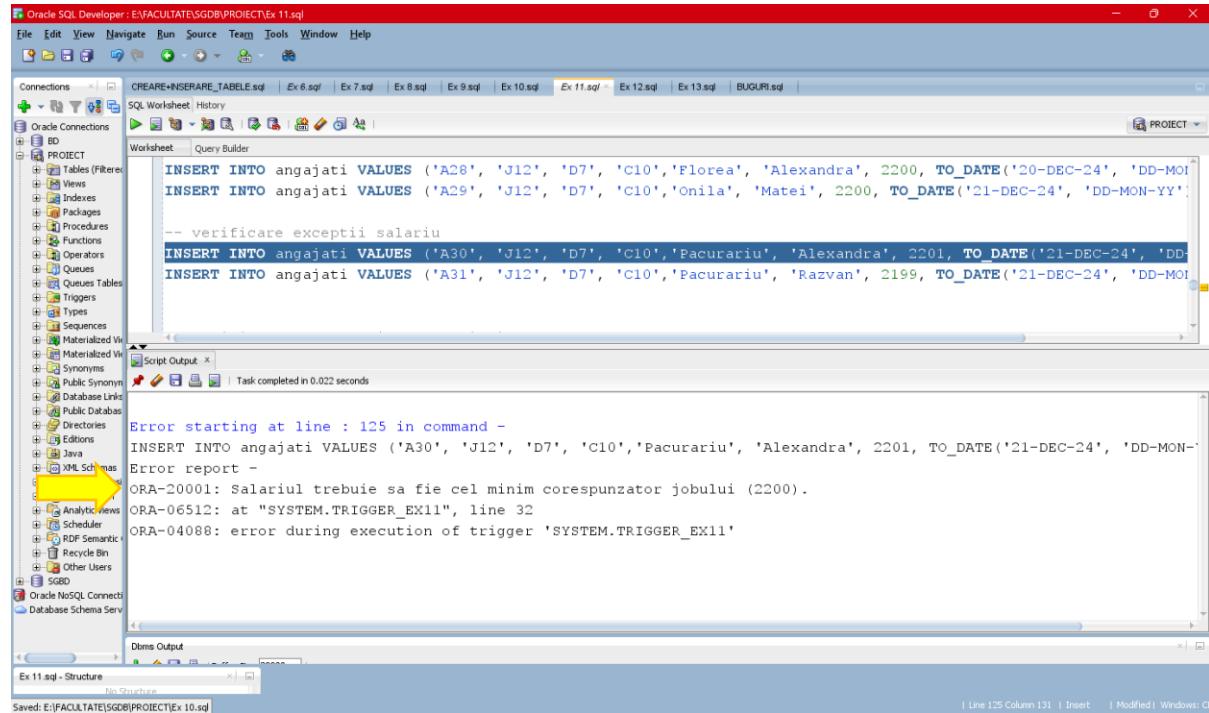
```

Cod angajat: A28
Operatie: Adaugare
Detalii modificarare: Angajat nou cu salariul minim corespunzator jobului (2200).
Data operatiei: 03-JAN-2025 17:00

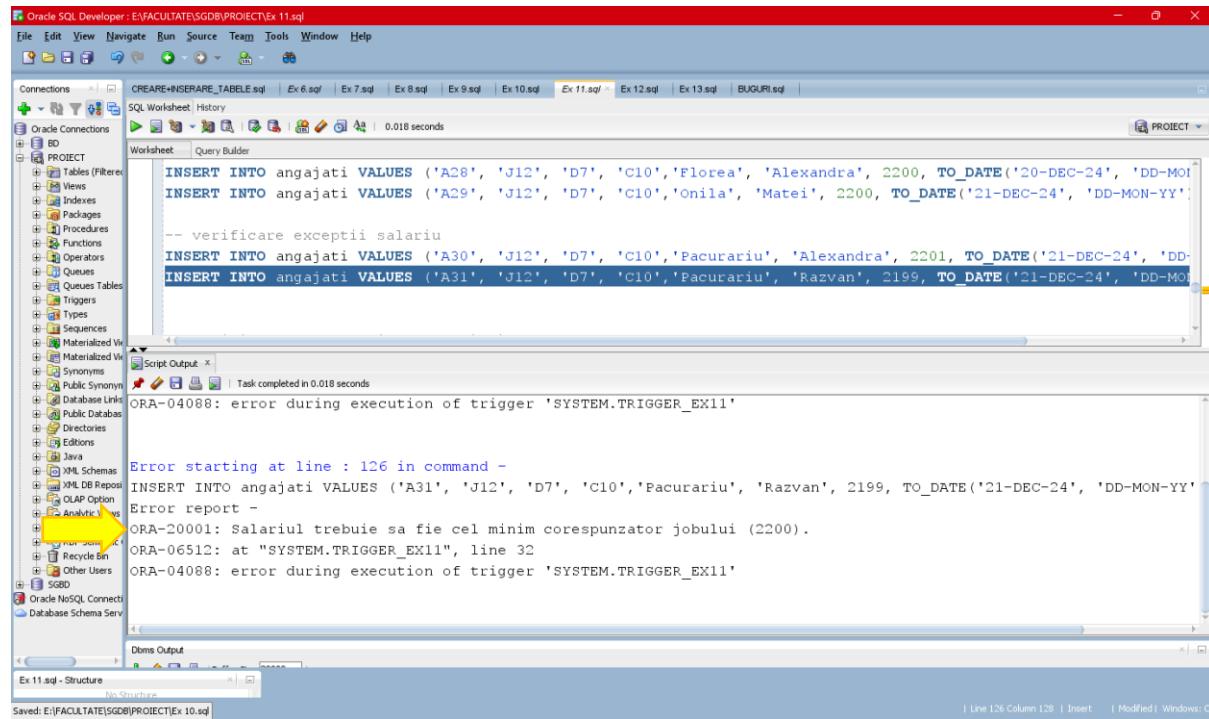
Cod angajat: A29
Operatie: Adaugare
Detalii modificarare: Angajat nou cu salariul minim corespunzator jobului (2200).
Data operatiei: 03-JAN-2025 17:00

-- verificare exceptii salariu

```
INSERT INTO angajati VALUES ('A30', 'J12', 'D7', 'C10','Pacurariu', 'Alexandra', 2201,  
TO_DATE('21-DEC-24', 'DD-MON-YY'),'Activ');  
INSERT INTO angajati VALUES ('A31', 'J12', 'D7', 'C10','Pacurariu', 'Razvan', 2199, TO_DATE('21-  
DEC-24', 'DD-MON-YY'),'Activ');
```



```
CREATE+INSERIRE_TABLE.sql | Ex 6.sql | Ex 7.sql | Ex 8.sql | Ex 9.sql | Ex 10.sql | Ex 11.sql | Ex 12.sql | Ex 13.sql | BUGURI.sql | PROJECT  
File Edit View Navigate Run Source Team Tools Window Help  
Connections BD PROJECT Tables (Filtered) Views Indexes Packages Procedures Functions Operators Queues Queues Tables Triggers Types Sequences Materialized Views Materialized Views Synonyms Public Synonyms Database Links Directories Editors Java XML Schemas XML Analytics XML DB Repos OLAP Option Recycle Bin Other Users SGBD Oracle NoSQL Connect Database Schema Serv  
Worksheet Query Builder  
INSERT INTO angajati VALUES ('A28', 'J12', 'D7', 'C10','Floarea', 'Alexandra', 2200, TO_DATE('20-DEC-24', 'DD-MON-YY'))  
INSERT INTO angajati VALUES ('A29', 'J12', 'D7', 'C10','Onila', 'Matei', 2200, TO_DATE('21-DEC-24', 'DD-MON-YY'))  
-- verificare exceptii salariu  
INSERT INTO angajati VALUES ('A30', 'J12', 'D7', 'C10','Pacurariu', 'Alexandra', 2201, TO_DATE('21-DEC-24', 'DD-MON-YY'))  
INSERT INTO angajati VALUES ('A31', 'J12', 'D7', 'C10','Pacurariu', 'Razvan', 2199, TO_DATE('21-DEC-24', 'DD-MON-YY'))  
  
Script Output x Task completed in 0.022 seconds  
Error starting at line : 125 in command -  
INSERT INTO angajati VALUES ('A30', 'J12', 'D7', 'C10','Pacurariu', 'Alexandra', 2201, TO_DATE('21-DEC-24', 'DD-MON-YY'))  
Error report -  
ORA-20001: Salariul trebuie sa fie cel minim corespunzator jobului (2200).  
ORA-06512: at "SYSTEM.TRIGGER_EX11", line 32  
ORA-04088: error during execution of trigger 'SYSTEM.TRIGGER_EX11'  
  
Dbs Output  
Ex 11.sql - Structure x|  
Saved: E:\FACULTATE\SGDB\PROJECT\Ex 10.sql | Line 125 Column 131 | Insert | Modified | Windows: C:\
```

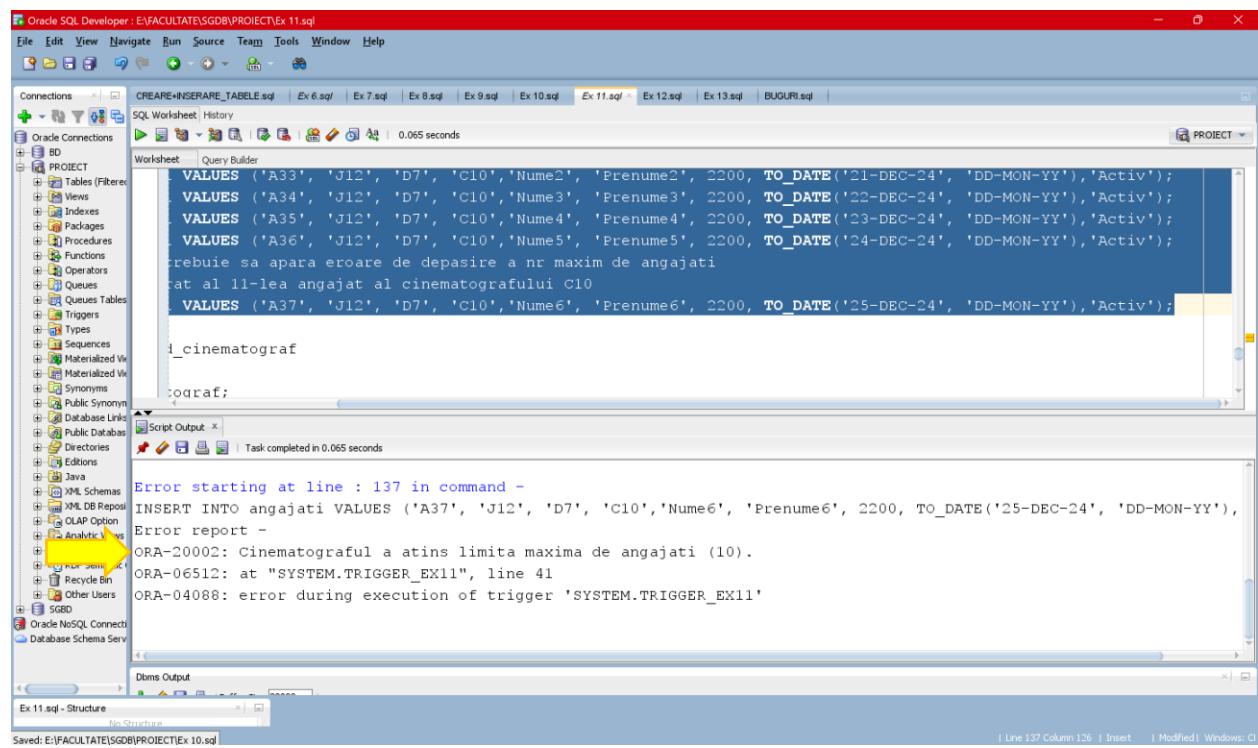


```
CREATE+INSERIRE_TABLE.sql | Ex 6.sql | Ex 7.sql | Ex 8.sql | Ex 9.sql | Ex 10.sql | Ex 11.sql | Ex 12.sql | Ex 13.sql | BUGURI.sql | PROJECT  
File Edit View Navigate Run Source Team Tools Window Help  
Connections BD PROJECT Tables (Filtered) Views Indexes Packages Procedures Functions Operators Queues Queues Tables Triggers Types Sequences Materialized Views Materialized Views Synonyms Public Synonyms Database Links Directories Editors Java XML Schemas XML DB Repos OLAP Option Recycle Bin Other Users SGBD Oracle NoSQL Connect Database Schema Serv  
Worksheet Query Builder  
INSERT INTO angajati VALUES ('A28', 'J12', 'D7', 'C10','Floarea', 'Alexandra', 2200, TO_DATE('20-DEC-24', 'DD-MON-YY'))  
INSERT INTO angajati VALUES ('A29', 'J12', 'D7', 'C10','Onila', 'Matei', 2200, TO_DATE('21-DEC-24', 'DD-MON-YY'))  
-- verificare exceptii salariu  
INSERT INTO angajati VALUES ('A30', 'J12', 'D7', 'C10','Pacurariu', 'Alexandra', 2201, TO_DATE('21-DEC-24', 'DD-MON-YY'))  
INSERT INTO angajati VALUES ('A31', 'J12', 'D7', 'C10','Pacurariu', 'Razvan', 2199, TO_DATE('21-DEC-24', 'DD-MON-YY'))  
  
Script Output x Task completed in 0.018 seconds  
ORA-04088: error during execution of trigger 'SYSTEM.TRIGGER_EX11'  
  
Error starting at line : 126 in command -  
INSERT INTO angajati VALUES ('A31', 'J12', 'D7', 'C10','Pacurariu', 'Razvan', 2199, TO_DATE('21-DEC-24', 'DD-MON-YY'))  
Error report -  
ORA-20001: Salariul trebuie sa fie cel minim corespunzator jobului (2200).  
ORA-06512: at "SYSTEM.TRIGGER_EX11", line 32  
ORA-04088: error during execution of trigger 'SYSTEM.TRIGGER_EX11'  
  
Dbs Output  
Ex 11.sql - Structure x|  
Saved: E:\FACULTATE\SGDB\PROJECT\Ex 10.sql | Line 126 Column 128 | Insert | Modified | Windows: C:\
```

```
-- verificare pt nr maxim de angajati
INSERT INTO angajati VALUES ('A32', 'J12', 'D7', 'C10','Nume1', 'Prenume1', 2200, TO_DATE('20-DEC-24', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A33', 'J12', 'D7', 'C10','Nume2', 'Prenume2', 2200, TO_DATE('21-DEC-24', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A34', 'J12', 'D7', 'C10','Nume3', 'Prenume3', 2200, TO_DATE('22-DEC-24', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A35', 'J12', 'D7', 'C10','Nume4', 'Prenume4', 2200, TO_DATE('23-DEC-24', 'DD-MON-YY'),'Activ');
INSERT INTO angajati VALUES ('A36', 'J12', 'D7', 'C10','Nume5', 'Prenume5', 2200, TO_DATE('24-DEC-24', 'DD-MON-YY'),'Activ');
```

-- la acest insert trebuie sa apara eroare de depasire a nr maxim de angajati
-- deoarece am inserat al 11-lea angajat al cinematografului C10

```
INSERT INTO angajati VALUES ('A37', 'J12', 'D7', 'C10','Nume6', 'Prenume6', 2200, TO_DATE('25-DEC-24', 'DD-MON-YY'),'Activ');
```



The screenshot shows the Oracle SQL Developer interface with the following details:

- Title Bar:** Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 11.sql
- Toolbar:** File, Edit, View, Navigate, Run, Source, Team, Tools, Window, Help.
- Connections:** Oracle Connections, BD, PROJECT.
- Project Explorer:** Tables (Filtered), Views, Indexes, Packages, Procedures, Functions, Operators, Queues, Queues Tables, Triggers, Types, Sequences, Materialized Views, Synonyms, Public Synonyms, Database Links, Public Databases, Directories, Editions, Java, XML Schemas, XML DB Repos, OLAP Option, Analytic View, Other Users, SGBD.
- SQL Worksheet:** History, Ex 6.sql, Ex 7.sql, Ex 8.sql, Ex 9.sql, Ex 10.sql, Ex 11.sql (highlighted), Ex 12.sql, Ex 13.sql, BUGURI.sql.
- Script Output:** Task completed in 0.065 seconds.
- Error Report:**

```
Error starting at line : 137 in command -
INSERT INTO angajati VALUES ('A37', 'J12', 'D7', 'C10','Nume6', 'Prenume6', 2200, TO_DATE('25-DEC-24', 'DD-MON-YY'),'Activ');

ORA-20002: Cinematograful a atins limita maxima de angajati (10).
ORA-06512: at "SYSTEM.TRIGGER_EX11", line 41
ORA-04088: error during execution of trigger 'SYSTEM.TRIGGER_EX11'
```
- Bottom Status Bar:** Line 137 Column 126 | Insert | Modified | Windows: C

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 11.sql

```

File Edit View Navigate Run Source Team Tools Window Help
Connections BD PROJECT
Tables (Filtered)
Views Indexes Packages Procedures Functions Operators Queues Queues Tables Triggers Types Sequences Materialized Views Materialized Views Synonyms Public Synonyms Database Links Public Databases Directories Editions Java XML Schemas XML DB Repos OLAP Option Analytic Views Scheduler RDF Semantic Recycle Bin Other Users SGBD Oracle NoSQL Connect Database Schema Serv
CREATE-INSERTARE_TABLE.sql Ex 6.sql Ex 7.sql Ex 8.sql Ex 9.sql Ex 10.sql Ex 11.sql Ex 12.sql Ex 13.sql BUGURI.sql
SQL Worksheet History
Worksheet Query Builder

```

INSERT INTO angajati VALUES ('A36', 'J12', 'D7', 'C10', 'Nume5', 'Pnume5', 2200, TO_DATE('24-DEC-24'), 'DD-MON-14')
-- la acest insert trebuie sa apara eroare de depasire a nr maxim de angajati
-- deoarece am inserat al 11-lea angajat al cinematografului C10

INSERT INTO angajati VALUES ('A37', 'J12', 'D7', 'C10', 'Nume6', 'Pnume6', 2200, TO_DATE('25-DEC-24'), 'DD-MON-14')

select count(*), cod_cinematograf
from angajati
group by cod_cinematograf;

delete from angajati

Script Output x Query Result x

All Rows Fetched: 10 in 0.005 seconds

COUNT(*)	COD_CINEMATOGRAF
1	2C8
2	3C6
3	1C5
4	5C1
5	10C10
6	3C2
7	3C3
8	2C9
9	3C4
10	2C7

Doms Output

Ex 11.sql - Structure No Structure

Saved: E:\FACULTATE\SGDB\PROJECT\Ex 10.sql

Line 141 Column 27 | Insert | Modified | Windows: C

-- verificare update
-- se va modifica angajatul 'A1'

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 11.sql

```

File Edit View Navigate Run Source Team Tools Window Help
Connections BD PROJECT
Tables (Filtered)
Views Indexes Packages Procedures Functions Operators Queues Queues Tables Triggers Types Sequences Materialized Views Materialized Views Synonyms Public Synonyms Database Links Public Databases Directories Editions Java XML Schemas XML DB Repos OLAP Option Analytic Views Scheduler RDF Semantic Recycle Bin Other Users SGBD Oracle NoSQL Connect Database Schema Serv
CREATE-INSERTARE_TABLE.sql Ex 6.sql Ex 7.sql Ex 8.sql Ex 9.sql Ex 10.sql Ex 11.sql Ex 12.sql Ex 13.sql BUGURI.sql
SQL Worksheet History
Worksheet Query Builder

```

from angajati
group by cod_cinematograf;

delete from angajati
where cod_angajat = 'A37';

select * from angajati;

--verificare update

Script Output x Query Result x

All Rows Fetched: 34 in 0.004 seconds

COD_ANGAJAT	COD_JOB	COD_DEPARTAMENT	COD_CINEMATOGRAF	NUME_ANGAJAT	PNUME_ANGAJAT	SALARIU_ANGAJAT	DATA_ANGAJARE	STATUS_ANGAJAT
1 A1	J1	D1	C1	Alexandrescu Andra	4500	16-JAN-23	Activ	
2 A2	J2	D2	C1	Roman Cristina	4500	20-FEB-23	Activ	
3 A3	J3	D3	C3	Pop Ioana	5500	25-MAR-23	In concediu	
4 A4	J4	D4	C4	Georgiu Ana	4050	30-APR-23	Activ	
5 A5	J5	D5	C1	Moldovan Roxana	3200	02-MAY-23	Activ	
6 A6	J6	D6	C6	David Andreea	3150	03-JUN-23	Suspendat	
7 A7	J7	D7	C7	Iacob Andrei	2300	18-JUL-19	Activ	
8 A8	J1	D1	C9	Vlonga Stefan	6000	20-AUG-23	Activ	
9 A9	J2	D2	C10	Durlesteau Victor	4000	25-SEP-23	In concediu	
10 A10	J3	D3	C1	Becchea Flavia	5000	24-OCT-19	Activ	
11 A11	J4	D4	C2	Racovita Cristina	6100	05-NOV-21	Activ	
12 A12	J5	D5	C8	-	-	-	-	

Doms Output

Ex 11.sql - Structure No Structure

Saved: E:\FACULTATE\SGDB\PROJECT\Ex 10.sql

Line 147 Column 24 | Insert | Modified | Windows: C

```
update angajati
set status_angajat = 'Suspendat'
where cod_angajat = 'A1';
```

The screenshot shows the Oracle SQL Developer interface with the following details:

- File Path:** E:\FACULTATE\SGDB\PROJECT\Ex 11.sql
- Connections:** BD
- Tables (Filtered):** angajati
- SQL Worksheet:**

```
delete from angajati
where cod_angajat = 'A37';

select * from angajati;

--verificare update
update angajati
set status_angajat = 'Suspendat'
where cod_angajat = 'A1';
```
- Script Output:** Task completed in 0.025 seconds
- Query Result:** 1 row updated.
- Logs:** Cod angajat: A1
Operatie: Suspendare
Detalii modificarie: Salariul a fost redus de la 4500 la 4275, dar nu sub salariul minim (4000).
Data operatiei: 03-JAN-2025 17:17

The screenshot shows the Oracle SQL Developer interface with the following details:

- File Path:** E:\FACULTATE\SGDB\PROJECT\Ex 11.sql
- Connections:** BD
- Tables (Filtered):** angajati
- SQL Worksheet:**

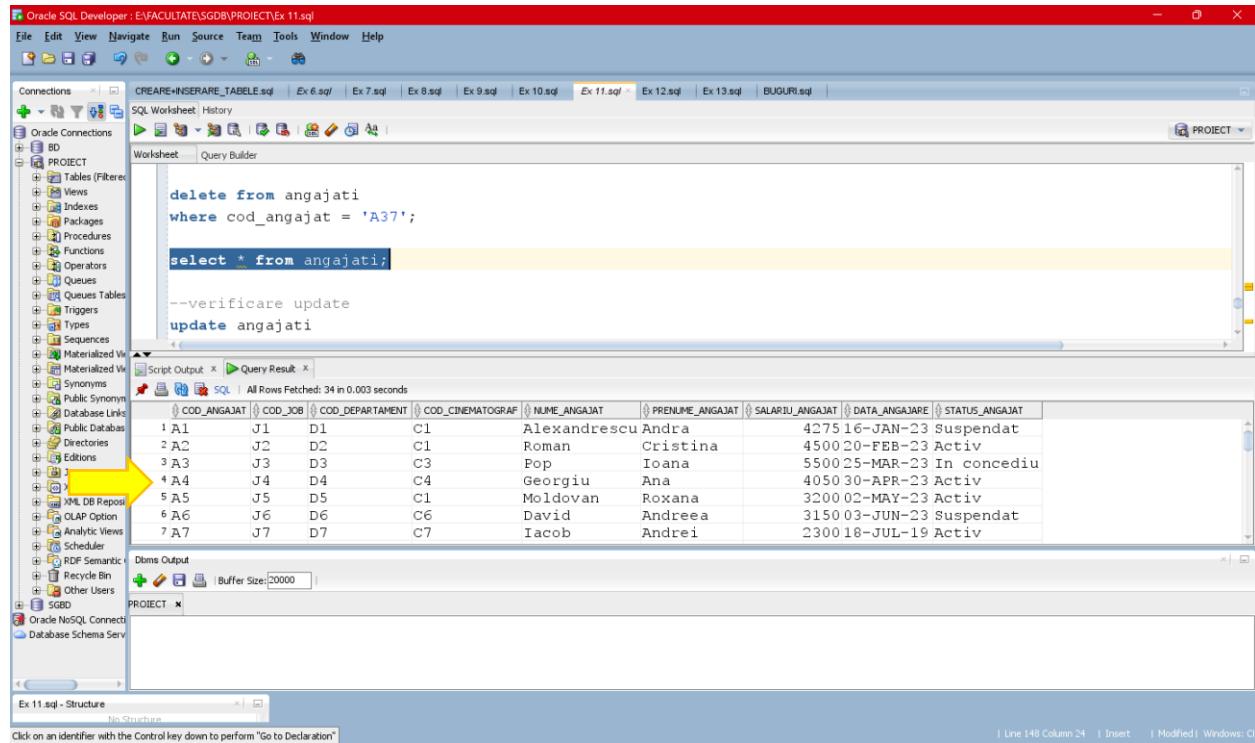
```
delete from angajati
where cod_angajat = 'A37';

select * from angajati;

--verificare update
update angajati
where cod_angajat = 'A1';
```
- Script Output:** All Rows Fetched: 34 in 0.003 seconds
- Query Result:**

COD_ANGAJAT	COD_JOB	COD_DEPARTAMENT	COD_CINEMATOGRAF	NUME_ANGAJAT	PRENUME_ANGAJAT	SALARIU_ANGAJAT	DATA_ANGAJARE	STATUS_ANGAJAT
1 A1	J1	D1	C1	Alexandrescu	Andra	4275	16-JAN-23	Suspendat
2 A2	J2	D2	C1	Roman	Cristina	4500	20-FEB-23	Activ
3 A3	J3	D3	C3	Pop	Ioana	5500	25-MAR-23	In concediu
4 A4	J4	D4	C4	Georgiu	Ana	4050	30-APR-23	Activ
5 A5	J5	D5	C1	Moldovan	Roxana	3200	02-MAY-23	Activ
- Logs:** Cod angajat: A1
Operatie: Suspendare
Detalii modificarie: Salariul a fost redus de la 4500 la 4275, dar nu sub salariul minim (4000).
Data operatiei: 03-JAN-2025 17:17

-- verificare update, dar reducerea de salariu este mai mica decat salariul minim
 -- se va modifica angajatul 'A4'



```

CREATE+INSERARE_TABLE.sql | Ex 6.sql | Ex 7.sql | Ex 8.sql | Ex 9.sql | Ex 10.sql | Ex 11.sql | Ex 12.sql | Ex 13.sql | BUGURI.sql |
File Edit View Navigate Run Source Team Tools Window Help
PROJECT
Tables (Filters)
Views
Indexes
Packages
Procedures
Functions
Operators
Queues
Queues Tables
Triggers
Types
Sequences
Materialized Views
Public Synonyms
Database Links
Public Databases
Directories
Editions
XML DB Repositories
OLAP Options
Analytic Views
Scheduler
RDF Semantic
Recycle Bin
Other Users
SGBD
Oracle NoSQL Connectors
Database Schema Service
Ex 11.sql - Structure
No Structure
Click on an identifier with the Control key down to perform "Go to Declaration"

```

```

delete from angajati
where cod_angajat = 'A37';

select * from angajati;

--verificare update
update angajati

```

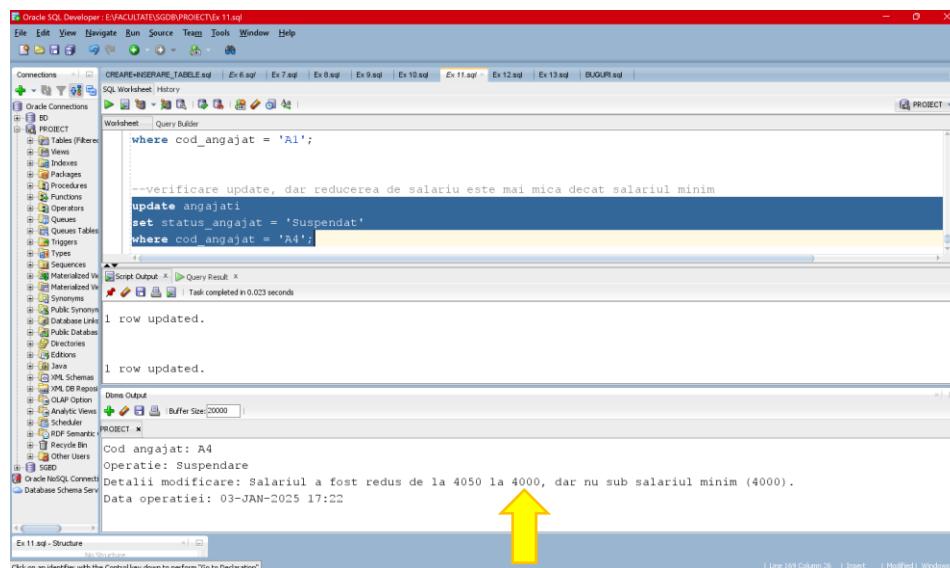
COD_ANGAJAT	COD_JOB	COD_DEPARTAMENT	COD_CINEMATOGRAF	NUME_ANGAJAT	PRENUME_ANGAJAT	SALARIU_ANGAJAT	DATA_ANGAJARE	STATUS_ANGAJAT
1_A1	J1	D1	C1	Alexandrescu	Andra	4275	16-JAN-23	Suspendat
2_A2	J2	D2	C1	Roman	Cristina	4500	20-FEB-23	Activ
3_A3	J3	D3	C3	Pop	Ioana	5500	25-MAR-23	In concediu
4_A4	J4	D4	C4	Georgiu	Ana	4050	30-APR-23	Activ
5_A5	J5	D5	C1	Moldovan	Roxana	3200	02-MAY-23	Activ
6_A6	J6	D6	C6	David	Andreea	3150	03-JUN-23	Suspendat
7_A7	J7	D7	C7	Iacob	Andrei	2300	18-JUL-19	Activ

```

update angajati
set status_angajat = 'Suspendat'
where cod_angajat = 'A4';

```

-- salariul ar fi fost redus de la 4050 la 3847 daca nu ar fi existat conditia ca reducerea salariului
 -- sa fie cel putin egala cu salariul minim al jobului ocupat



```

CREATE+INSERARE_TABLE.sql | Ex 6.sql | Ex 7.sql | Ex 8.sql | Ex 9.sql | Ex 10.sql | Ex 11.sql | Ex 12.sql | Ex 13.sql | BUGURI.sql |
File Edit View Navigate Run Source Team Tools Window Help
PROJECT
Tables (Filters)
Views
Indexes
Packages
Procedures
Functions
Operators
Queues
Queues Tables
Triggers
Types
Sequences
Materialized Views
Public Synonyms
Database Links
Public Databases
Directories
Editions
XML DB Repositories
OLAP Options
Analytic Views
Scheduler
RDF Semantic
Recycle Bin
Other Users
SGBD
Oracle NoSQL Connectors
Database Schema Service
Ex 11.sql - Structure
No Structure
Click on an identifier with the Control key down to perform "Go to Declaration"

```

```

where cod_angajat = 'A1';

--verificare update, dar reducerea de salariu este mai mica decat salariul minim
update angajati
set status_angajat = 'Suspendat'
where cod_angajat = 'A4';

```

1 row updated.

1 row updated.

Cod angajat: A4
 Operatie: Suspendare
 Detalii modificarare: Salariu a fost redus de la 4050 la 4000, dar nu sub salariul minim (4000).
 Data operatiei: 03-JAN-2025 17:22

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 11.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections Oracle Connections BD Oracle Tables Filtered Views Indexes Procedures Functions Operators Queues Triggers Types Sequences Materialized Views Synonyms Public Synonyms Database Links Public Databases Directories XML Schemas XML DB Repos OLAP Option Analytic Views Scheduler RDF Semantic Recycle Bin Other Users Oracle NoSQL Connect Database Schema Serv

Worksheet Query Builder

```

CREATE+INSERARE_TABLEB.sql | Ex 6.sql | Ex 7.sql | Ex 8.sql | Ex 9.sql | Ex 10.sql | Ex 11.sql | Ex 12.sql | Ex 13.sql | BUGURI.sql | PROJECT

```

delete from angajati
where cod_angajat = 'A37';

select * from angajati;

--verificare update
update angajati

Script Output | Query Result

COD_ANGAJAT	COD_JOB	COD_DEPARTAMENT	COD_CINEMATOGRAF	NUME_ANGAJAT	PRENUME_ANGAJAT	SALARIU_AN	DATA_ANGAJARE	SITUA_ANGAJAT
1 A1	J1	D1	C1	Alexandrescu Andra		4000	16-JAN-23	Suspendat
2 A2	J2	D2	C1	Roman Cristina		5500	20-FEB-23	
3 A3	J3	D3	C3	Pop Ioana		5500	25-MAR-23	In concediu
4 A4	J4	D4	C4	Georgiu Ana		4000	30-APR-23	Suspendat
5 A5	J5	D5	C1	Moldovan Roxana		3200	02-MAY-23	Activ

Dims Output

PROJECT x

Cod angajat: A4
Operatie: Suspendare
Detalii modificarile: Salariul a fost redus de la 4050 la 4000, dar nu sub salariul minim (4000).
Data operatiei: 03-JAN-2025 17:22

Ex 11.sql - Structure

No Structure

Click on an identifier with the Control key down to perform "Go to Declaration"

Line 148 Column 24 | Insert | Modified | Windows: C

-- primele 10 inserari sunt relevante acestui exercitiu

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 11.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections AQ\$_QUEUE_TABLES AQ\$_QUEUES AQ\$_SCHEDULES AUDIT_USER CINEMATOGRAP CLIENT DEPARTAMENTE DIFUZEAZA FILM HELP INFO_ANGAJATI ISTORIC_ERORI JOBURI LOGMN\$_AGE_SPILLS LOGMN\$_ATTRCOLS LOGMN\$_ATTRIBUTES LOGMN\$_CCLS LOGMN\$_CDFS LOGMN\$_COL\$ LOGMN\$_COLTYPE\$ LOGMN\$_CON LOGMN\$_CONTAINER LOGMN\$_DICTIONARY LOGMN\$_DICTSTATE\$ LOGMN\$_DIDS LOGMN\$_ENC\$ LOGMN\$_ERROR\$ LOGMN\$_GT_TAB_INK LOGMN\$_GT_USER_IN LOGMN\$_ICL\$ LOGMN\$_JONSEQ\$ LOGMN\$_IND\$ LOGMN\$_INDCOMPAR LOGMN\$_INDPARTS LOGMN\$_TUNING_PARAMETERS

Worksheet Query Builder

```

CREATE+INSERARE_TABLEB.sql | Ex 6.sql | Ex 7.sql | Ex 8.sql | Ex 9.sql | Ex 10.sql | Ex 11.sql | Ex 12.sql | Ex 13.sql | BUGURI.sql | PROJECT

```

113
114 select *
115 from angajati;
116
117 select * from info_angajati;

Script Output | Query Result

ID_INFO	COD	OPERATIE	DETALII_COLOANA_MODIFICATA	DATA_INFO
1	1A28	Adaugare	Angajat nou cu salariul minim corespunzator jobului (2200).	03-JAN-25
2	2A29	Adaugare	Angajat nou cu salariul minim corespunzator jobului (2200).	03-JAN-25
3	3A32	Adaugare	Angajat nou cu salariul minim corespunzator jobului (2200).	03-JAN-25
4	4A33	Adaugare	Angajat nou cu salariul minim corespunzator jobului (2200).	03-JAN-25
5	5A34	Adaugare	Angajat nou cu salariul minim corespunzator jobului (2200).	03-JAN-25
6	6A35	Adaugare	Angajat nou cu salariul minim corespunzator jobului (2200).	03-JAN-25
7	7A36	Adaugare	Angajat nou cu salariul minim corespunzator jobului (2200).	03-JAN-25
8	8A1	Suspendare	Salariul a fost redus de la 4275, dar nu sub salariul minim (4000).	03-JAN-25
9	9A1	Suspendare	Salariul a fost redus de la 4500 la 4275, dar nu sub salariul minim (4000).	03-JAN-25
10	10A4	Suspendare	Salariul a fost redus de la 4050 la 4000, dar nu sub salariul minim (4000).	03-JAN-25
11	21A29	Adaugare	Angajat nou cu salariul minim corespunzator jobului (2200).	04-JAN-25

Dims Output

PROJECT x

Ex 11.sql - Structure

No Structure

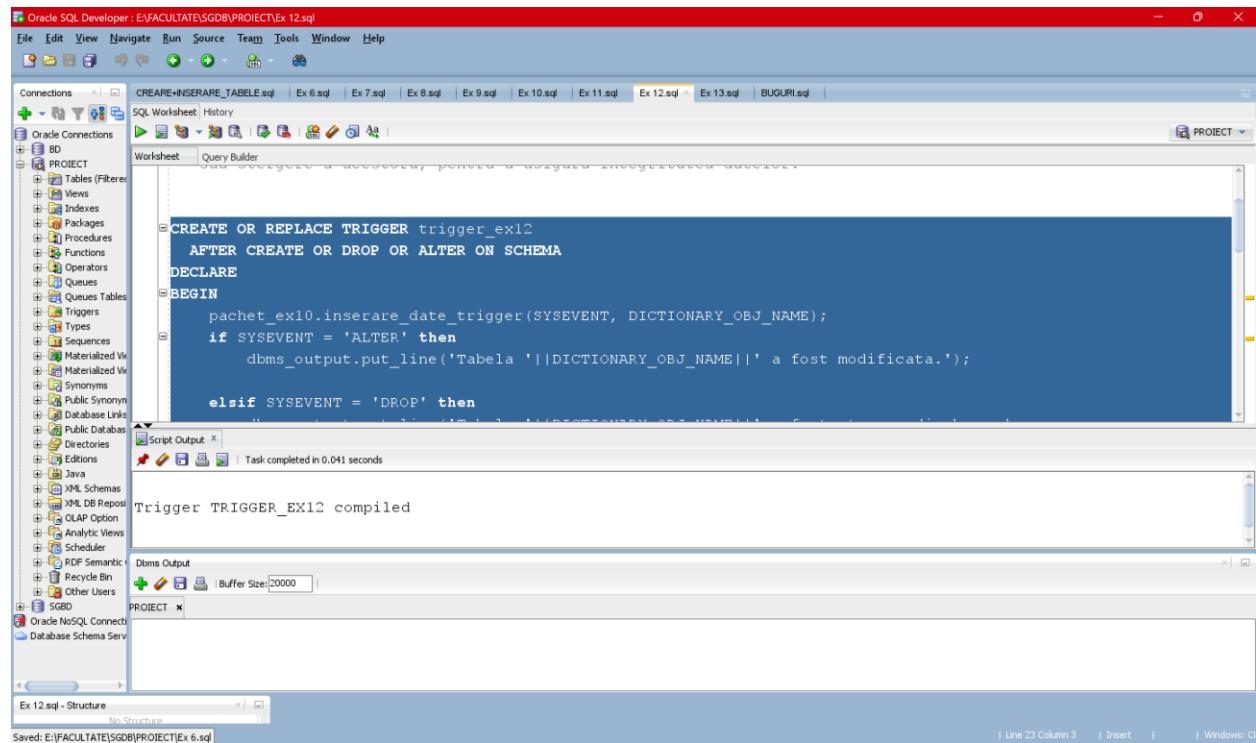
Saved: E:\FACULTATE\SGDB\PROJECT\Ex 10.sql

Line 117 Column 29 | Insert | Modified | Windows: C

12. Trigger de tip LDD

Având în vedere importanța fiecărui tabel din baza de date a unui cinematograf, se solicită înregistrarea detaliată a tuturor operațiunilor de creare, modificare sau ștergere a acestora, pentru a asigura integritatea datelor.

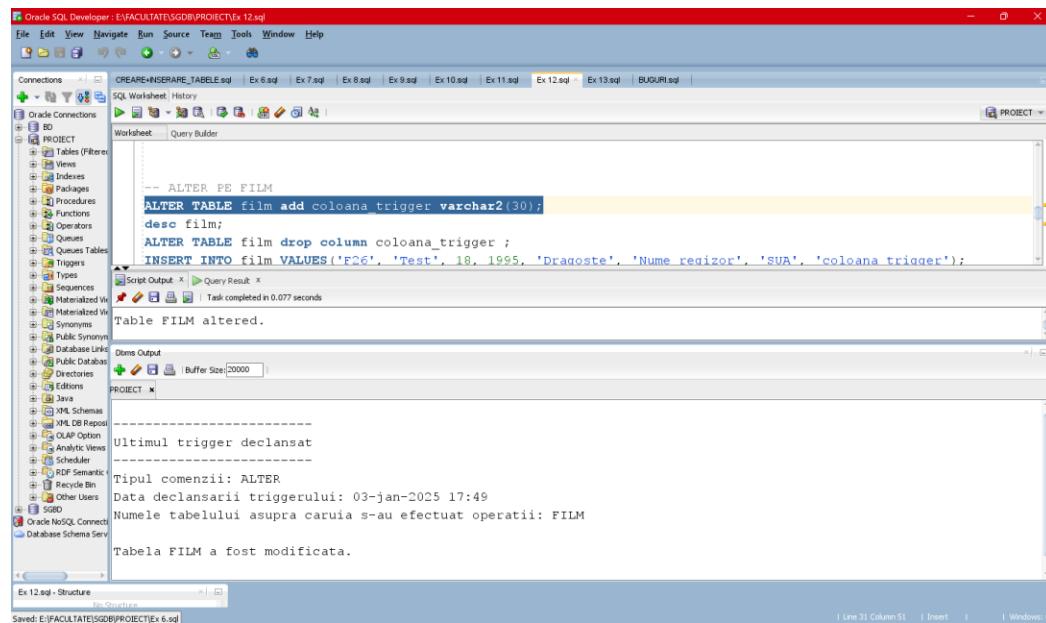
```
CREATE OR REPLACE TRIGGER trigger_ex12
AFTER CREATE OR DROP OR ALTER ON SCHEMA
DECLARE
BEGIN
    pachet_ex10.inserare_date_trigger(SYSEVENT, DICTIONARY_OBJ_NAME);
    if SYSEVENT = 'ALTER' then
        dbms_output.put_line('Tabela '||DICTIONARY_OBJ_NAME||' a fost modificata.');
    elsif SYSEVENT = 'DROP' then
        dbms_output.put_line('Tabela '||DICTIONARY_OBJ_NAME||' a fost stearsa din baza de
                           date');
    elsif SYSEVENT = 'CREATE' then
        dbms_output.put_line('Tabela '||DICTIONARY_OBJ_NAME||' a fost creata.');
    end if;
END;
/
```



Testare trigger_ex12

-- ALTER PE FILM

```
ALTER TABLE film add coloana_trigger varchar2(30);
desc film;
ALTER TABLE film drop column coloana_trigger ;
INSERT INTO film VALUES('F26', 'Test', 18, 1995, 'Dragoste', 'Nume regizor', 'SUA',
'coloana_trigger');
```



```
-- ALTER PE FILM
ALTER TABLE film add coloana_trigger varchar2(30);
desc film;
ALTER TABLE film drop column coloana_trigger ;
INSERT INTO film VALUES('F26', 'Test', 18, 1995, 'Dragoste', 'Nume regizor', 'SUA',
'coloana_trigger');

Table FILM altered.
```

Doms Output

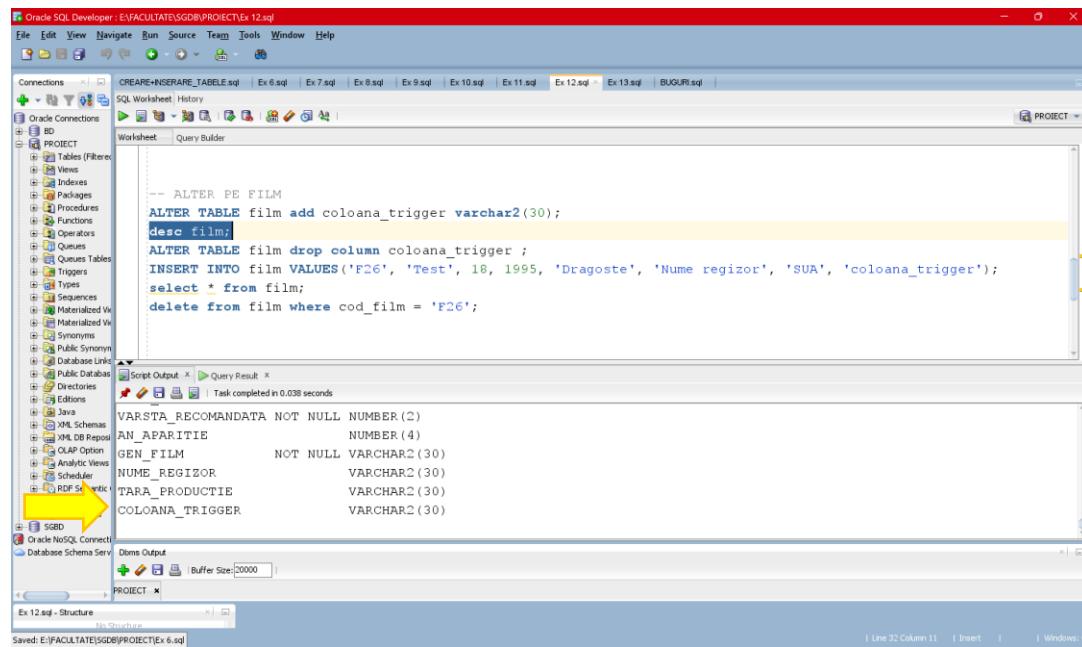
Ultimul trigger declarat

Tipul comenzii: ALTER

Data declansarii triggerului: 03-jan-2025 17:49

Numele tabelului asupra caruia s-au efectuat operatii: FILM

Tabela FILM a fost modifitata.



```
-- ALTER PE FILM
ALTER TABLE film add coloana_trigger varchar2(30);
desc film;
ALTER TABLE film drop column coloana_trigger ;
INSERT INTO film VALUES('F26', 'Test', 18, 1995, 'Dragoste', 'Nume regizor', 'SUA',
'coloana_trigger');
select * from film;
delete from film where cod_film = 'F26';

VARTA_RECOMANDATA NOT NULL NUMBER(2)
AN_APARITIE          NUMBER(4)
GEN_FILM              NOT NULL VARCHAR2(30)
NUME_REGIZOR          VARCHAR2(30)
TARA_PRODUCTIE        VARCHAR2(30)
COLOANA_TRIGGER        VARCHAR2(30)
```

-- ALTER PE FILM
ALTER TABLE film add coloana_trigger varchar2(30);
desc film;
ALTER TABLE film drop column coloana_trigger ;
INSERT INTO film VALUES('F26', 'Test', 18, 1995, 'Dragoste', 'Nume regizor', 'SUA', 'coloana_trigger');

Table FILM altered.

Script Output | Query Result | Task completed in 0.117 seconds

Ultimul trigger declansat

Tipul comenzii: ALTER
Data declansarii triggerului: 03-jan-2025 17:50
Numele tabelului asupra caruia s-au efectuat operatii: FILM
Tabela FILM a fost modificata.

Ex 12.sql - Structure | No Structure | Line 33 Column 47 | Insert | Windows: C

Saved: E:\FACULTATE\SGDB\PROJECT\Ex 6.sql

-- deoarece am eliminat coloana 'coloana_trigger', ne va aparea urmatoarea eroare

-- ALTER PE FILM
ALTER TABLE film add coloana_trigger varchar2(30);
desc film;
ALTER TABLE film drop column coloana_trigger ;
INSERT INTO film VALUES('F26', 'Test', 18, 1995, 'Dragoste', 'Nume regizor', 'SUA', 'coloana_trigger');
select * from film;
delete from film where cod_film = 'F26';

Error at Command Line : 34 Column : 13
Error report -
SQL Error: ORA-00913: too many values
00913. 00000 - "too many values"
*Cause:
*Action:

Doms Output | Buffer Size:20000 | Line 34 Column 104 | Insert | Windows: C

Ex 12.sql - Structure | No Structure | Line 34 Column 104 | Insert | Windows: C

Saved: E:\FACULTATE\SGDB\PROJECT\Ex 6.sql

```
-- CREATE, ALTER, DROP PE O TABELA CREATĂ CA TEST
CREATE TABLE test_trigger(
    coloana1 varchar2(30),
    coloana2 varchar2(30),
    numar number);
ALTER TABLE test_trigger add coloana4 varchar2(30);
desc test_trigger;
INSERT INTO test_trigger VALUES('Test', 'trigger_ex12', 12, 'ultima_coloana');
select * from test_trigger;
DROP TABLE test_trigger;
```

The screenshot shows the Oracle SQL Developer interface. The 'Connections' sidebar is open, showing a connection to 'BD'. The 'PROJECT' node is expanded, showing 'Tables (Filtered)' which contains 'test_trigger'. The 'Worksheet' tab is active, displaying the SQL code for creating the table. The 'Script Output' tab shows the message 'Table TEST_TRIGGER created.' The 'Query Result' tab shows the output of the 'SELECT *' query, listing the four columns: coloana1, coloana2, numar, and coloana4.

The screenshot shows the Oracle SQL Developer interface again. The 'Connections' sidebar is open, showing a connection to 'BD'. The 'PROJECT' node is expanded, showing 'Tables (Filtered)' which contains 'test_trigger'. The 'Worksheet' tab is active, displaying the SQL code for altering the table to add a new column. The 'Script Output' tab shows the message 'Table TEST_TRIGGER altered.' The 'Query Result' tab shows the output of the 'SELECT *' query, listing the five columns: coloana1, coloana2, numar, coloana4, and the newly added column.

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 12.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections Oracle Connections BD PROJECT

Tables (Filtered) Views Indexes Packages Procedures Functions Operators Queues Queues Tables Triggers Types Sequences Materialized Views Materialized Views Synonyms Public Synonyms Database Links Public Databases Directories Editions Java XML Schemas XML DB Repos OLAP Option Analytic Views Scheduler RDF Semantic Recycle Bin Oracle NoSQL Connect Database Schema Serv

Worksheet Query Builder

```

CREATE+INSERTAR_TABLE.sql | Ex 6.sql | Ex 7.sql | Ex 8.sql | Ex 9.sql | Ex 10.sql | Ex 11.sql | Ex 12.sql | Ex 13.sql | BUGURI.sql | PROJECT
```

```

coloanal varchar2(30),
coloana2 varchar2(30),
numar number);
ALTER TABLE test_trigger add coloana4 varchar2(30);
desc test trigger;
INSERT INTO test_trigger VALUES('Test', 'trigger_ex12', 12, 'ultima_coloana');
select * from test_trigger;
DROP TABLE test_trigger;
```

Script Output | Query Result | Task completed in 0.603 seconds

Table TEST_TRIGGER altered.

Name	Null?	Type
COLOANA1		VARCHAR2 (30)
COLOANA2		VARCHAR2 (30)
NUMAR		NUMBER
COLOANA4		VARCHAR2 (30)

Dims Output | Buffer Size: 20000 | PROJECT x

Ex 12.sql - Structure | No Structure | Saved: E:\FACULTATE\SGDB\PROJECT\Ex 6.sql | Line 46 Column 19 | Insert | Windows: C

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 12.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections Oracle Connections BD PROJECT

Tables (Filtered) Views Indexes Packages Procedures Functions Operators Queues Queues Tables Triggers Types Sequences Materialized Views Materialized Views Synonyms Public Databases Directories Editions Java XML Schemas XML DB Repos OLAP Option Analytic Views Scheduler RDF Semantic Recycle Bin Oracle NoSQL Connect Database Schema Serv

Worksheet Query Builder

```

CREATE+INSERTAR_TABLE.sql | Ex 6.sql | Ex 7.sql | Ex 8.sql | Ex 9.sql | Ex 10.sql | Ex 11.sql | Ex 12.sql | Ex 13.sql | BUGURI.sql | PROJECT
```

```

coloanal varchar2(30),
coloana2 varchar2(30),
numar number);
ALTER TABLE test_trigger add coloana4 varchar2(30);
desc test trigger;
INSERT INTO test_trigger VALUES('Test', 'trigger_ex12', 12, 'ultima_coloana');
```

Script Output | Query Result | Task completed in 0.029 seconds

COLOANA4 VARCHAR2 (30)

1 row inserted.

Dims Output | Buffer Size: 20000 | PROJECT x

UIMUL trigger declansat

Tipul comenzi: ALTER

Data declansarii triggerului: 03-jan-2025 17:57

Numele tabelului asupra caruia s-au efectuat operatii: TEST_TRIGGER

Tabela TEST_TRIGGER a fost modifitata.

Ex 12.sql - Structure | No Structure | Saved: E:\FACULTATE\SGDB\PROJECT\Ex 6.sql | Line 47 Column 79 | Insert | Windows: C

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 12.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections BD PROJECT

SQL Worksheet: History Worksheet Query Builder

```
ALTER TABLE test_trigger add coloana4 varchar2(30);
desc test_trigger;
INSERT INTO test_trigger VALUES('Test', 'trigger_ex12', 12, 'ultima_coloana');
select * from test_trigger;
DROP TABLE test_trigger;
```

Script Output X Query Result X

All Rows Fetched: 1 in 0.003 seconds

COLOANA1	COLOANA2	NUMAR	COLOANA4
1	Test	trigger ex12	12ultima coloana

Doms Output

+ Buffer Size:20000

PROJECT x

Uitmul trigger declansat

Tipul comenzii: ALTER

Data declansarii triggerului: 03-jan-2025 17:57

Numele tabelului asupra caruia s-au efectuat operatii: TEST_TRIGGER

Tabela TEST_TRIGGER a fost modificata.

Ex 12.sql - Structure No Structure

Saved: E:\FACULTATE\SGDB\PROJECT\Ex 6.sql

Line 48 Column 28 Insert Windows: C

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 12.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections BD PROJECT

SQL Worksheet: History Worksheet Query Builder

```
ALTER TABLE test_trigger add coloana4 varchar2(30);
desc test_trigger;
INSERT INTO test_trigger VALUES('Test', 'trigger_ex12', 12, 'ultima_coloana');
select * from test_trigger;
DROP TABLE test_trigger;
```

Script Output X Query Result X

Task completed in 0.057 seconds

Table TEST_TRIGGER dropped.

Doms Output

+ Buffer Size:20000

PROJECT x

Ultimul trigger declansat

Tipul comenzii: DROP

Data declansarii triggerului: 03-jan-2025 17:59

Numele tabelului asupra caruia s-au efectuat operatii: TEST_TRIGGER

Tabela TEST_TRIGGER a fost stearsa din baza de date

Ex 12.sql - Structure No Structure

Saved: E:\FACULTATE\SGDB\PROJECT\Ex 6.sql

Line 49 Column 25 Insert Windows: C

13. Pachet complex (se utilizează DBMS_JOB)

Cu prilejul Zilei Naționale a României, cinematografele din întreaga țară au organizat o serie de evenimente speciale pentru a celebra această zi. Evenimentele includ:

1. Tarife unice pentru bilete, indiferent de categoria de vîrstă:

- *Film 2D: 10 lei*
- *Film 3D: 15 lei*
- *Film IMAX: 20 lei*

2. Surprize cinematografice speciale:

a. Clasament pe baza recenziilor: S-a realizat un clasament general al filmelor, indiferent de compania de cinematografe. Filmul cel mai apreciat a fost adăugat în programul zilei de 1 decembrie 2024.

b. Filme speciale:

- *Lansare în avanpremieră: Red One (aventură de Crăciun).*
- *Beetlejuice (film clasic de Halloween) rămâne în program inclusiv pe 1 decembrie.*
- *Anul Nou care n-a fost (regizat de Bogdan Mureșanu) pentru celebrarea Zilei Naționale.*

3. Surprize pentru clienți:

a. La finalul fiecărei zile, clientul cu cele mai multe puncte bonus va primi un bilet gratuit pentru un film la alegere. Notificarea va fi trimisă clientului la ora 22:00.

b. Fiecare client care a făcut rezervare pe 1 decembrie a fost notificat să acorde o recenzie pentru filmul vizionat.

S-a realizat un clasament al filmelor, care include:

- *Numărul de rezervări atribuite fiecărui film.*
- *Recenziile lăsate de clienți.*

Clasamentul astfel realizat va fi folosit pentru a ține cont de preferințele clientilor în planificarea ofertelor pentru anul următor.

```
-- secventa necesara in momentul adaugarii recenziilor noi in baza de date
CREATE SEQUENCE recenzi_seq
START WITH 32
INCREMENT BY 1;
```

The screenshot shows the Oracle SQL Developer interface. In the central workspace, there is a query window titled 'Query Builder' containing the following SQL code:

```
31
32
33
34
35 -- secventa necesara in momentul adaugarii recenziilor noi in baza de date
36 CREATE SEQUENCE recenzi_seq
37 START WITH 32
38 INCREMENT BY 1;
39
40
41 -- In momentul in care un client face o rezervare, acest trigger actualizeaza tabela
42 -- CLIENTI cu punctele bonus obtinute.
43 -- Se va afisa de asemenea un mesaj de atentionare legat de restrictia de varsta a filmului.
```

Below the code, the message 'Sequence RECENTII_SEQ created.' is displayed. The bottom status bar indicates the file is saved at 'E:\FACULTATE\SGDB\PROJECT\Ex 13.sql'.

```
-- In momentul in care un client face o rezervare, acest trigger actualizeaza tabela
-- CLIENTI cu punctele bonus obtinute.
-- Se va afisa de asemenea un mesaj de atentionare legat de restrictia de varsta a filmului.
```

```
CREATE OR REPLACE TRIGGER actualizare_puncte_bonus
```

```
FOR INSERT ON rezervare
```

```
COMPOUND TRIGGER
```

```
puncte_1_decembrie number := 0;
data_1_decembrie date := TO_DATE('01-12-2024', 'DD-MM-YYYY');
data_gasita boolean := false;
type vector is varray(10) of date;
date_difuzare vector := vector();
varsta_film.varsta_recomandata%type;
```

```
cursor c_date is
select t.data_difuzare
from difuzeaza d, table(d.perioade_difuzare) t
where d.cod_film = :NEW.cod_film;
```

```
BEFORE EACH ROW IS
```

```
BEGIN
```

```

open c_date;
fetch c_date bulk collect into date_difuzare;
close c_date;

for i in date_difuzare.first..date_difuzare.last loop
    if date_difuzare(i) = data_1_decembrie then
        data_gasita := true;
        exit;
    end if;
end loop;

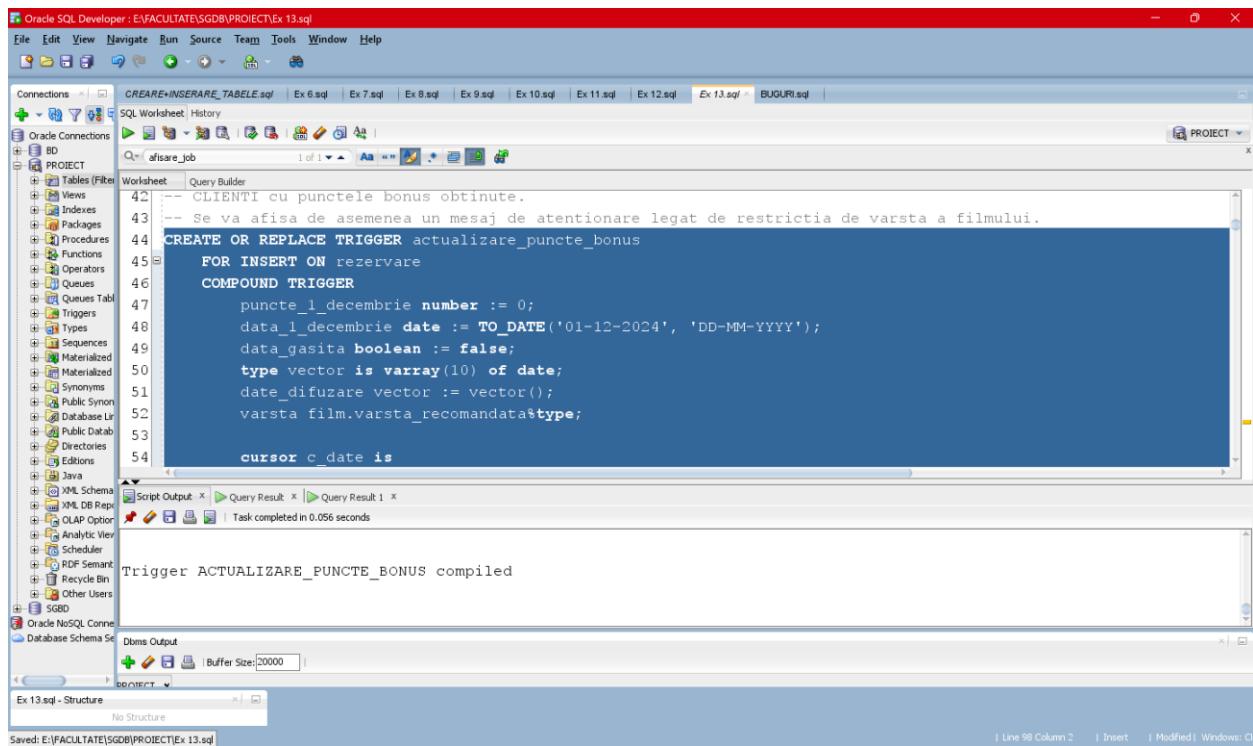
if data_gasita = true then
    if :NEW.format_proiectie = '2D' then
        puncte_1_decembrie := 10;
    elsif :NEW.format_proiectie = '3D' then
        puncte_1_decembrie := 15;
    elsif :NEW.format_proiectie = 'IMAX' then
        puncte_1_decembrie := 20;
    end if;
else
    dbms_output.put_line('Rezervarea nu este facuta pentru un film din date de 1
decembrie, punctele bonus vor fi adaugate fara adaosul special');
end if;
END BEFORE EACH ROW;

AFTER EACH ROW IS
BEGIN
    update clienti
    set puncte_bonus = puncte_bonus+
puncte_1_decembrie+:NEW.pret_bilet*:NEW.numar_persoane
    where cod_client = :NEW.cod_client;

    select varsta_recomandata into varsta
    from film
    where cod_film = :NEW.cod_film;

    dbms_output.put_line('Rezervarea a fost facuta cu succes!');
    dbms_output.put_line('Aveti in vedere ca varsta minima la acest film este: '||varsta|| ' ani.');
END AFTER EACH ROW;
END actualizare_puncte_bonus;
/

```



```

CREATE OR REPLACE PACKAGE pachet_ex13 AS
  type record_top is record(
    cod_film.cod_film%type,
    nume_film.nume_film%type,
    regizor_film.nume_regizor%type,
    numar_rez number,
    nota_medie number
  );
  type tablou_imbricat is table of record_top;
  type vector_clienti is varray(20) of clienti.cod_client%type;
  type vector_cod_filme is varray(20) of film.cod_film%type;
  type vector_nume_filme is table of film.nume_film%type index by pls_integer;

  function client_cu_max_puncte_bonus return number;
  function clienti_care_dau_recenzii return vector_clienti;
  function topul_filmelor_coduri return tablou_imbricat;

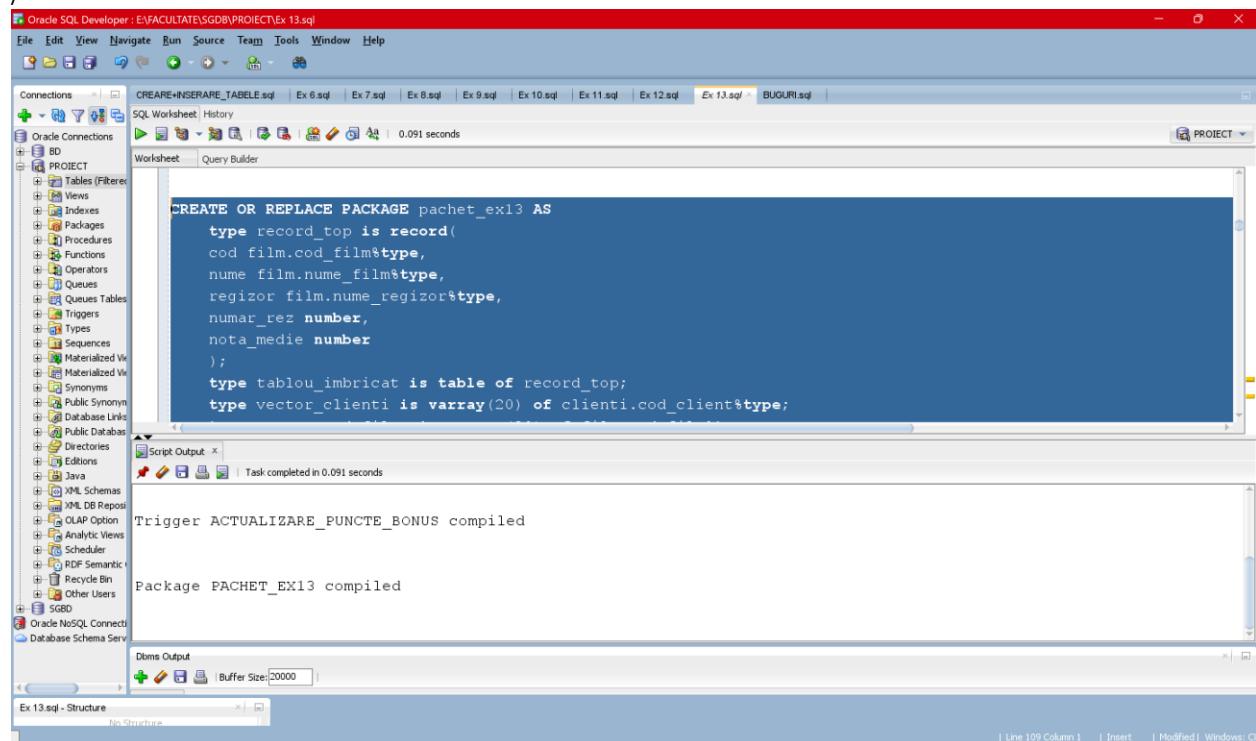
  procedure adauga_film_rezervare;
  procedure notif_max_pct_bonus;
  procedure submit_notif_max_pct_bonus_job;
  procedure notif_recenzii;

```

```

procedure submit_notif_recenzii;
procedure adauga_recenzii(v_clienti in vector_clienti);
procedure afisare_top_filme(v_recorduri in tablou_imbricat);
END pachet_ex13;
/

```



```
CREATE OR REPLACE PACKAGE BODY pachet_ex13 AS
```

----- TOPUL FILMELOR - functie -----

```

function topul_filmelor_coduri return tablou_imbricat
is
    v_recorduri tablou_imbricat := tablou_imbricat();
    cursor c_top_filme is
        select f.cod_film, f.nume_film, f.nume_regizor,
               nvl(count(rez.cod_film), 0) nr_rezervari,
               nvl(round(avg(rec.scor),2), 0) nota_mediie
        from film f
            join rezervare rez on f.cod_film = rez.cod_film
            join clienti c on rez.cod_client = c.cod_client
            join recenzie rec on c.cod_client = rec.cod_client
        group by f.cod_film, f.nume_film, f.nume_regizor
        order by nr_rezervari desc, nota_mediie desc;

```

```

begin
  open c_top_filme;
  fetch c_top_filme bulk collect into v_recorduri;
  close c_top_filme;

  return v_recorduri;
end topul_filmelor_coduri;

----- ADAUGA FILME, REZERVARI, DIFUZARI- procedura -----
procedure adauga_film_rezervare
is
  v_top tablou_imbricat := topul_filmelor_coduri;
  primul_din_top record_top := v_top(v_top.first);
  id_cinema varchar2(3);
  durata varchar2(3);
begin
  select cod_cinematograf, durata_film
  into id_cinema, durata
  from difuzeaza
  where cod_film = primul_din_top.cod;

  INSERT INTO TABLE (
    select d.perioade_difuzare
    from difuzeaza d
    where d.cod_cinematograf = id_cinema
      and d.cod_film = primul_din_top.cod)
  select per_dif(
    TO_DATE('01-12-2024', 'DD-MM-YYYY'),
    TO_DATE('18:00', 'HH24:MI'),
    TO_DATE('18:00', 'HH24:MI') + durata / 1440
  ) FROM dual;
  INSERT INTO rezervare VALUES (primul_din_top.cod, 8, 10, 3, '2D', 4, 'Card');

  INSERT INTO film VALUES('F26', 'Red one', 8, 2024, 'Actiune', 'Jake Kasdan', 'SUA');
  INSERT INTO difuzeaza VALUES (
    'C4', 'F26',
    lista_perioade(
      per_dif(TO_DATE('01-12-2024', 'DD-MM-YYYY'), TO_DATE('16:30', 'HH24:MI'),
      TO_DATE('19:00', 'HH24:MI'))),
    'Engleza', 120);
  INSERT INTO rezervare VALUES ('F26', 10, 10, 2, '2D', 1, 'Card');
  INSERT INTO rezervare VALUES ('F26', 11, 10, 4, '2D', 1, 'Cash');
  INSERT INTO rezervare VALUES ('F26', 12, 10, 3, '2D', 1, 'Cash');

```

```

INSERT INTO film VALUES('F27', 'BeetleJuice', 12, 2024, 'Comedie', 'Tim Burton', 'SUA');
INSERT INTO difuzeaza VALUES (
    'C4', 'F27',
    lista_perioade(
        per_dif(TO_DATE('01-12-2024', 'DD-MM-YYYY'), TO_DATE('14:30', 'HH24:MI'),
        TO_DATE('16:00', 'HH24:MI'))),
    'Romana', 90);
INSERT INTO rezervare VALUES ('F27', 22, 20, 2, 'IMAX', 2, 'Card');
INSERT INTO rezervare VALUES ('F27', 7, 20, 2, 'IMAX', 2, 'Cash');

INSERT INTO film VALUES('F28', 'Anul Nou Care N-a Fost', 15, 2024, 'Drama', 'Bogdan
Muresanu', 'Romania');
INSERT INTO difuzeaza VALUES (
    'C4', 'F28',
    lista_perioade(
        per_dif(TO_DATE('01-12-2024', 'DD-MM-YYYY'), TO_DATE('20:00', 'HH24:MI'),
        TO_DATE('22:00', 'HH24:MI'))),
    'Romana', 120);
INSERT INTO rezervare VALUES ('F28', 21, 15, 1, '3D', 3, 'Card');
INSERT INTO rezervare VALUES ('F28', 4, 15, 4, '3D', 3, 'Cash');
INSERT INTO rezervare VALUES ('F28', 5, 15, 3, '3D', 3, 'Card');
INSERT INTO rezervare VALUES ('F28', 17, 15, 3, '3D', 3, 'Cash');
end adauga_film_rezervare;

```

----- CLIENT MAX PCT BONUS - functie -----

```

function client_cu_max_puncte_bonus return number
is
    cod_client_cu_max_pct number;
    max_puncte number := 0;
    max_rezervari number := 0;
    nr_rezervari_client number;
begin
    for i in (select puncte_bonus, cod_client
               from clienti ) loop
        if i.puncte_bonus > max_puncte then
            cod_client_cu_max_pct := i.cod_client;
            max_puncte := i.puncte_bonus;

            select count(*) into max_rezervari
            from rezervare
            where cod_client = i.cod_client;

```

```

-- daca exista mai multi clienti cu acelasi nr de puncte bonus se va determina clientul
-- care a facut mai multe rezervari
elsif i.puncte_bonus = max_puncte then
    select count(*) into nr_rezervari_client
    from rezervare
    where cod_client = i.cod_client;

    if nr_rezervari_client > max_rezervari then
        max_rezervari := nr_rezervari_client;
        cod_client_cu_max_pct := i.cod_client;
    end if;
end if;
end loop;
return cod_client_cu_max_pct;
end client_cu_max_puncte_bonus;

```

----- NOTIF PCT BONUS-procedura -----

```

procedure notif_max_pct_bonus is
    nume varchar2(30);
    prenume varchar2(30);
begin
    select nume_client, prenume_client
    into nume, prenume
    from clienti
    where cod_client = client_cu_max_puncte_bonus;

    dbms_output.put_line('Notificare trimisa catre: '||nume||' '||prenume);
    dbms_output.put_line('Felicitari, ai castigat o vizionare gratuita din cabina de proiectie a
oricarui film ales de tine!');
    dbms_output.put_line('Oferta este valabila pe perioada a doua saptamani incepand cu data
la care ai primit mesajul.');
end notif_max_pct_bonus;

```

----- DBMS_JOB PENTRU PCT BONUS - procedura -----

```

procedure submit_notif_max_pct_bonus_job is
    nr_job_id number;
begin
    DBMS_JOB.SUBMIT(
        JOB      => nr_job_id,
        WHAT     => 'pachet_ex13.notif_max_pct_bonus;',
        NEXT_DATE => TRUNC(SYSDATE) + 22/24,
        INTERVAL  => 'TRUNC(SYSDATE + 1) + 22/24' );
    dbms_output.put_line('Job-ul cu ID-ul '||nr_job_id||' a fost inregistrat cu succes.');

```

```

EXCEPTION
WHEN OTHERS THEN
    RAISE_APPLICATION_ERROR(-20050,'Eroare la înregistrarea job-ului: ' || SQLERRM);
end submit_notif_max_pct_bonus_job;

```

----- DETERMINARE CLIENTI CARE TREBUIE SA DEA RECENZII - functie -----

```

function clienti_care_dau_recenzii return vector_clienti
is
    v_clienti vector_clienti := vector_clienti();
    cursor c_clienti is
        select c.cod_client
        from clienti c
        join rezervare r on c.cod_client = r.cod_client
        join film f on r.cod_film = f.cod_film
        join difuzeaza d on f.cod_film = d.cod_film, table(d.perioade_difuzare) t
        where t.data_difuzare = TO_DATE('01-12-2024', 'DD-MM-YYYY')
        group by c.cod_client
        order by c.cod_client;
begin
    open c_clienti;
    fetch c_clienti bulk collect into v_clienti;
    close c_clienti;

    return v_clienti;
end clienti_care_dau_recenzii;

```

----- NOTIFICARE RECENZII - procedura -----

```

procedure notif_recenzii
is
    v_clienti vector_clienti := clienti_care_dau_recenzii;
    v_nume_filme vector_nume_filme;
    nume clienti.nume_client%type;
    prenume clienti.prenume_client%type;

    -- se selecteaza cele 4 filme care s-au difuzat pe 1 decembrie
    cursor c_nume_filme(cod clienti.cod_client%type) is
        select f.nume_film nume
        from rezervare r
        join film f on r.cod_film = f.cod_film
        join difuzeaza d on f.cod_film = d.cod_film, table(d.perioade_difuzare) t
        where t.data_difuzare = TO_DATE('01-12-2024', 'DD-MM-YYYY')
        and r.cod_client = cod;

```

```

begin
  for i in v_clienti.first..v_clienti.last loop
    v_nume_filme := vector_nume_filme();

    for j in c_nume_filme(v_clienti(i)) loop
      v_nume_filme(v_nume_filme.count + 1) := j.nume;
    end loop;

    select nume_client, prenume_client
    into nume, prenume
    from clienti
    where cod_client = v_clienti(i);

    dbms_output.put_line('Notificare trimisa catre '||nume||' ||| prenume||| pentru
urmatoarele filme vizionate:');
    for k in v_nume_filme.first..v_nume_filme.last loop
      dbms_output.put_line(' '||v_nume_filme(k));
    end loop;
    dbms_output.put_line('Ne bucuram ca ati ales sa va petreceti ziua de 1 decembrie la
cinematograf!');
    dbms_output.put_line('Va rugam lasati o recenzie filmelor pe care ati ales sa le vizionati.');
    dbms_output.new_line;
  end loop;
end notif_recenzii;

```

----- DBMS_JOB PENTRU RECENZII - procedura -----

```

procedure submit_notif_recenzii is
  v_job_id number;
begin
  DBMS_JOB.SUBMIT(
  JOB => v_job_id,
  WHAT => 'pachet_ex13.notif_recenzii',
  NEXT_DATE => SYSDATE+3/86400,
  INTERVAL => 'SYSDATE+1'
  );

  dbms_output.put_line('Job-ul cu ID-ul '||v_job_id||' a fost inregistrat cu succes.');
EXCEPTION
  WHEN OTHERS THEN
    RAISE_APPLICATION_ERROR(-20051,'Eroare la înregistrarea job-ului: ' || SQLERRM);
end submit_notif_recenzii;

```

----- INSERARE RECENZII - procedura -----

```
procedure adauga_recenzii(v_clienti in vector_clienti)
is
    v_cod_filme vector_cod_filme;
    v_nota number;
    v_id_recenzie varchar2(3);

    cursor c_cod_filme(cod clienti.cod_client%type) is
        select f.cod_film cod
        from rezervare r
            join film f on r.cod_film = f.cod_film
            join difuzeaza d on f.cod_film = d.cod_film, table(d.perioade_difuzare) t
            where t.data_difuzare = TO_DATE('01-12-2024', 'DD-MM-YYYY')
            and r.cod_client = cod;
begin
    for i in v_clienti.first..v_clienti.last loop
        v_cod_filme := vector_cod_filme();

        for j in c_cod_filme(v_clienti(i)) loop
            v_cod_filme.extend;
            v_cod_filme(v_cod_filme.count) := j.cod;
        end loop;

        -- recenziile vor avea o nota random
        -- codul recenziei se construieste utilizand recenziile_seq definita anterior
        for k in v_cod_filme.first..v_cod_filme.last loop
            v_nota := ROUND(DBMS_RANDOM.VALUE(1,10));
            v_id_recenzie := 'R' || TO_CHAR(recenziile_seq.NEXTVAL);
            INSERT INTO recenzie VALUES (v_id_recenzie, v_cod_filme(k), v_clienti(i), v_nota,
TO_DATE('2024-12-02', 'YYYY-MM-DD'));
        end loop;
    end loop;
    dbms_output.put_line('Recenziile au fost adaugate cu succes');
end adauga_recenzii;
```

----- TOPUL FILMELOR - procedura -----

```
procedure afisare_top_filme(v_recorduri in tablou_imbricat) is
    -- in v_recorduri se retineau informatiile filmelor ordonate descrescator dupa popularitate
begin
    dbms_output.put_line('Topul filmelor din anul 2024 difuzate in cinematografe: ');
    for i in v_recorduri.first..v_recorduri.last loop
        dbms_output.put_line('Cod: ' || v_recorduri(i).cod);
        dbms_output.put_line('Nume film: ' || v_recorduri(i).nume);
        dbms_output.put_line('Regizor: ' || v_recorduri(i).regizor);
```

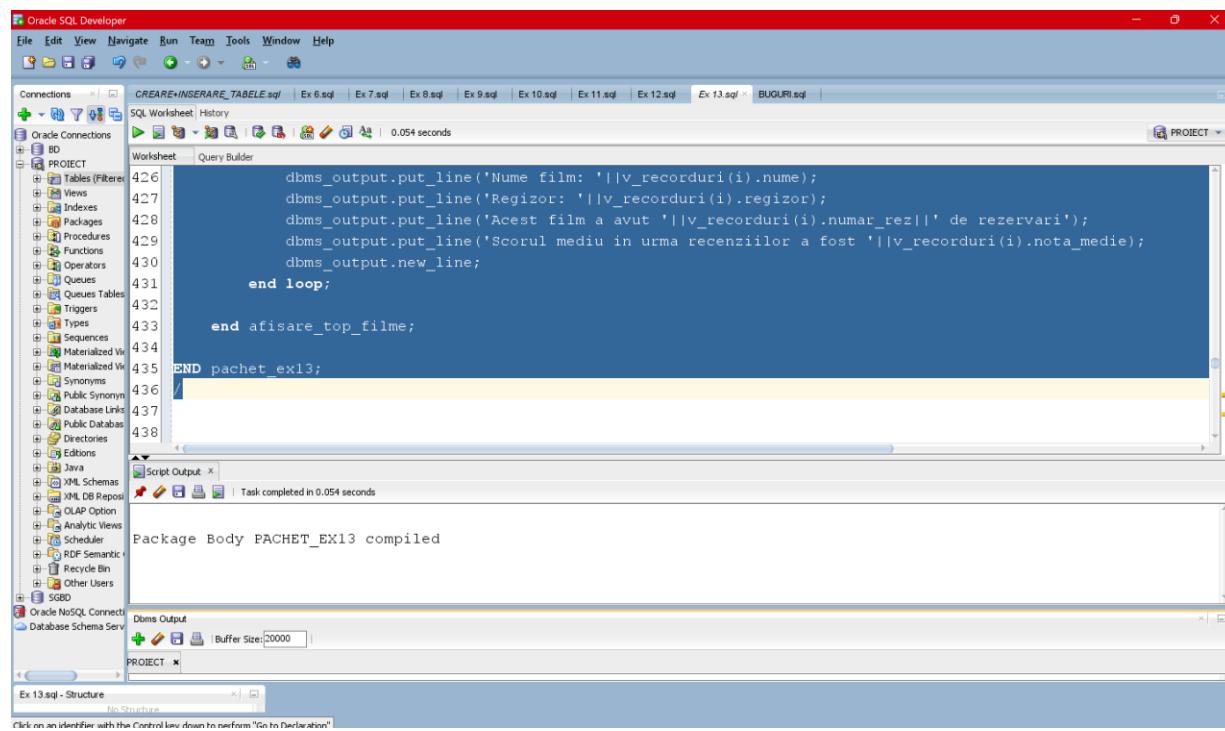
```

        dbms_output.put_line('Acest film a avut '||v_recorduri(i).numar_rez||' de rezervari');
        dbms_output.put_line('Scorul mediu in urma recenziilor a fost
    '||v_recorduri(i).nota_medie);
        dbms_output.new_line;
    end loop;

end afisare_top_filme;

END pachet_ex13;
/

```



Testare pachet_ex13

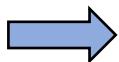
-- In continuare fiecare pas va fi compilat pe rand pentru a se putea verifica mai usor
-- functionalitatea pachetului.

Pasul 1

```

declare
    cod_client_max_pct_bonus number;
    v_clienti pachet_ex13.vector_clienti;
    v_coduri_filme pachet_ex13.tablou_imbricat;
begin

```



```
-- PASUL 1: se adauga date in tabelele FILM, DIFUZEAZA, REZERVARI
-- si se activeaza triggerul pt rezervari (se modif implicit si pct bonus din tabela CLIENTI)
pachet_ex13.adauga_film_rezervare;

--PASUL 2: determinarea clientului cu max pct bonus
--cod_client_max_pct_bonus := pachet_ex13.client_cu_max_puncte_bonus;
--dbms_output.put_line('Clientul avand codul '||cod_client_max_pct_bonus|||
--' a obtinut numar maxim de puncta.');

--PASUL 3: notificarea pentru clientul cu max pct bonus
--pachet_ex13.submit_notif_max_pct_bonus_job;

-- PASUL 4: notificare pentru acordarea de recenzii
--pachet_ex13.submit_notif_recenzii;

-- PASUL 5: adaugarea recenziilor
--v_clienti := pachet_ex13.clienti_care_dau_recenzii;
--pachet_ex13.adauga_recenzi(v_clienti);

-- PASUL 6: afisarea clasamentului final al filmelor
--v_coduri_filme := pachet_ex13.topul_filmelor_coduri;
--pachet_ex13.afisare_top_filme(v_coduri_filme);
end;
/
```

```
CREATE+INSERARE_TABLE.sq| Ex 6.sql | Ex 7.sql | Ex 8.sql | Ex 9.sql | Ex 10.sql | Ex 11.sql | Ex 12.sql | Ex 13.sql | BUGURI.sql | PROJECT
```

```
PL/SQL procedure successfully completed.
```

```
Tabela PACHET_EX13 a fost creata.
```

```
Rezervarea a fost facuta cu succes!
```

```
Aveti in vedere ca varsta minima la acest film este: 6 ani.
```

```
Rezervarea a fost facuta cu succes!
```

```
Aveti in vedere ca varsta minima la acest film este: 8 ani.
```

```
Rezervarea a fost facuta cu succes!
```

```
Aveti in vedere ca varsta minima la acest film este: 8 ani.
```

```
Rezervarea a fost facuta cu succes!
```

```
Aveti in vedere ca varsta minima la acest film este: 8 ani.
```

```
-- verificare PASUL 1
select * from film;
select * from rezervare;
select * from clienti; -- pentru verificarea triggerului
SELECT d.cod_cinematograf,
       d.cod_film,
       TO_CHAR(p.data_difuzare, 'DD-MM-YYYY') AS data_difuzare,
       TO_CHAR(p.ora_inceput, 'HH24:MI') AS ora_inceput,
       TO_CHAR(p.ora_final, 'HH24:MI') AS ora_final
FROM difuzeaza d,
     TABLE(d.perioade_difuzare) p;
```

-- se observa faptul ca in tabelul FILM s-au adaugat cele 3 filme noi: "Red One", "BeetleJuice" si "Anul Nou Care N-a Fost"

-- filmul cel mai apreciat nu se mai adauga in tabel deoarece este deja inserat

The screenshot shows the Oracle SQL Developer interface. In the top-left corner, the title bar reads "Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 13.sql". The menu bar includes File, Edit, View, Navigate, Run, Source, Team, Tools, Window, Help. The toolbar has various icons for connection management, schema navigation, and code editing.

The central workspace contains a "Worksheet" tab with the following SQL code:

```
466 end;
467 /
468
469 -- verificare PASUL 1
470 select * from film;
471 select * from rezervare;
472 select * from clienti; -- pentru verificarea triggerului
```

Below the code is a "Script Output" tab showing the results of the query:

COD_FILM	NUME_FILM	VARSTA_RECOMANDATA	AN_APARITIE	GEN_FILM	NUME_REGIZOR	TARA_PRODUCERIE
18	F18 Crouching Tiger, Hidden Dragon	12	2000	Actiune	Ang Lee	China
19	F19 The Grand Budapest Hotel	12	2014	Comedie	Wes Anderson	SUA
20	F20 Black Panther	13	2018	Supereroi	Ryan Coogler	SUA
21	F21 A Separation	12	2011	Drama	Asghar Farhadi	Iran
22	F22 Life is Beautiful	10	1997	Drama	Roberto Benigni	Italia
23	F23 Pan's Labyrinth	15	2006	Fantezie	Guillermo del Toro	Spania
24	F24 The Revenant	16	2015	Aventura	Alejandro González Iñárritu	SUA
25	F25 Amélie	12	2001	Comedie Romantica	Jean-Pierre Jeunet	Franta
26	F26 Red one	8	2024	Actiune	Jake Kasdan	SUA
27	F27 BeetleJuice	12	2024	Comedie	Tim Burton	SUA
28	F28 Anul Nou Care N-a Fost	15	2024	Drama	Bogdan Muresanu	Romania

At the bottom of the interface, there are tabs for "Doms Output" and "PROJECT". The status bar at the bottom right shows "Line 470 Column 20 | Insert | Modified | Windows: Cl".

-- initial in tabelul REZERVARE erau 33 de inserari, acum sunt 43.

-- 9 rezervari sunt facute la filmele noi adaugate si o rezervare este pentru filmul cel mai apreciat

The screenshot shows the Oracle SQL Developer interface. In the center is a 'Query Result' grid displaying data from a query. The columns are labeled: COD_FILM, COD_CLIENT, PRET_BILET, NUMAR_PERSOANE, FORMAT_PROIECTIE, NUMAR_SALA, and METODA_PLATA. The data consists of 15 rows, with the last row (F28) highlighted in yellow. A yellow arrow points from the left margin towards this highlighted row. The grid shows various movie codes like F20, F17, F26, etc., and client codes like 22, 8, 10, 11, 12, 20, 7, 21, 4, 5, 17. The 'Format Proiectie' column includes values like 4 2D, 3 2D, 2 2D, 4 2D, 3 2D, 2 IMAX, 2 IMAX, 1 3D, 4 3D, 3 3D, and 3 3D. The 'Metoda Plata' column includes values like 2 Card, 4 Card, 1 Card, 1 Cash, 1 Cash, 2 Card, 2 Cash, 3 Card, 3 Cash, and 3 Cash.

-- exemple de clienti a caror puncte bonus s-au modifical datorita triggerului:

Cod_client = 4

 INAINTE: 110 puncte_bonus, DUPA: 185 puncte_bonus

Cod_client = 5

 INAINTE: 175 puncte_bonus, DUPA: 235 puncte_bonus

Cod_client = 7

 INAINTE: 215 puncte_bonus, DUPA: 275 puncte_bonus

Cod_client = 10

 INAINTE: 135 puncte_bonus, DUPA: 165 puncte_bonus

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 13.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections BD PROJECT

Worksheet Query Builder

```

469 -- verificare PASUL 1
470 select * from film;
471 select * from rezervare;
472 select * from clienti; -- pentru verificarea triggerului
473 SELECT d.cod_cinematograf,
474     d.cod_film,
475     TO_CHAR(p.data_difuzare, 'DD-MM-YYYY') AS data_difuzare,

```

Script Output | Query Result | SQL | All Rows Fetched: 22 in 0.007 seconds

COD_CLIENT	NUME_CLIENT	PRENUME_CLIENT	PUNCTE_BONUS
1	Popescu	Ion	125
2	Ioanitoaiei	Maria	145
3	Georgian	Andrei	190
4	Marin	Ana	185
5	Dumitru	Mihai	235
6	Jalba	Elena	102
7	Popa	Florin	275
8	Stan	Ioana	225
9	Diaconu	Cristian	198
10	Radu	Vasile	165
11	Tudor	Monica	280
12			100
13			150
14			200
15			250
16			300
17			350
18			400
19			450
20			500
21			550
22			600
23			650
24			700
25			750
26			800
27			850
28			900
29			950
30			1000
31			1050
32			1100
33			1150
34			1200
35			1250
36			1300
37			1350
38			1400
39			1450
40			1500
41			1550
42			1600
43			1650
44			1700
45			1750
46			1800
47			1850
48			1900
49			1950
50			2000

Dbs Output | PROJECT | Buffer Size: 20000 | Line 472 Column 57 | Insert | Modified | Windows: C:\

Ex 13.sql - Structure | No Structure | Click on an identifier with the Control key down to perform "Go to Declaration"

-- inainte erau 46 de date la care se difuzau filmele, acum sunt 50, cate o difuzare in plus pentru cele 4 filme

Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex 13.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections BD PROJECT

Worksheet Query Builder

```

471 select * from rezervare;
472 select * from clienti; -- pentru verificarea triggerului
473 SELECT d.cod_cinematograf,
474     d.cod_film,
475     TO_CHAR(p.data_difuzare, 'DD-MM-YYYY') AS data_difuzare,
476     TO_CHAR(p.ora_inceput, 'HH24:MI') AS ora_inceput,

```

Script Output | Query Result | SQL | All Rows Fetched: 50 in 0.011 seconds

COD_CINEMATOGRAF	COD_FILM	DATA_DIFUZARE	ORA_INCEPUT	ORA_FINAL
38 C7	F19	06-12-2024 18:00	20:10	
39 C7	F19	07-12-2024 17:30	19:40	
40 C8	F20	19-12-2024 16:00	18:20	
41 C8	F20	02-12-2024 18:30	20:50	
42 C9	F21	10-12-2024 18:00	20:00	
43 C10	F22	23-12-2024 17:00	18:10	
44 C1	F23	15-12-2024 19:00	20:50	
45 C2	F24	27-12-2024 16:00	17:30	
46 C2	F24	02-12-2024 18:00	19:30	
47 C3	F25	10-12-2024 15:30	17:10	
48 C4	F26	01-12-2024 16:30	19:00	
49 C4	F27	01-12-2024 14:30	16:00	
50 C4	F28	01-12-2024 20:00	22:00	

Dbs Output | PROJECT | Buffer Size: 20000 | Line 479 Column 35 | Insert | Modified | Windows: C:\

Ex 13.sql - Structure | No Structure | Click on an identifier with the Control key down to perform "Go to Declaration"

Pasul 2

```
declare
    cod_client_max_pct_bonus number;
    v_clienti pachet_ex13.vector_clienti;
    v_coduri_filme pachet_ex13.tablou_imbricat;
begin
    -- PASUL 1: se adauga date in tabelele FILM, DIFUZEAZA, REZERVARI
    -- + se activeaza triggerul pt rezervari (se modif implicit si pct bonus din tabela CLIENTI)
    --pachet_ex13.adauga_film_rezervare;
```

→ **--PASUL 2: determinarea clientului cu max pct bonus**

```
cod_client_max_pct_bonus := pachet_ex13.client_cu_max_puncte_bonus;
dbms_output.put_line('Clientul avand codul '||cod_client_max_pct_bonus|||
    ' a obtinut numar maxim de puncte.');
```

--PASUL 3: notificarea pentru clientul cu max pct bonus

```
--pachet_ex13.submit_notif_max_pct_bonus_job;
```

-- PASUL 4: notificare pentru acordarea de recenzii

```
--pachet_ex13.submit_notif_recenzii;
```

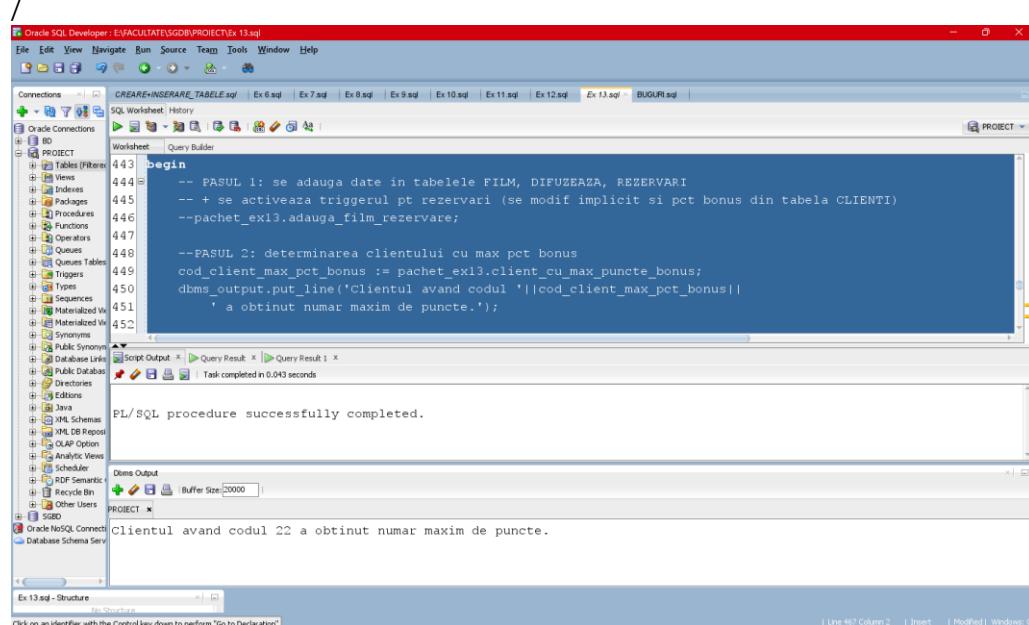
-- PASUL 5: adaugarea recenziilor

```
--v_clienti := pachet_ex13.clienti_care_dau_recenzii;
--pachet_ex13.adauga_recenzi(v_clienti);
```

-- PASUL 6: afisarea clasamentului final al filmelor

```
--v_coduri_filme := pachet_ex13.topul_filmelor_coduri;
--pachet_ex13.afisare_top_filme(v_coduri_filme);
```

```
end;
/
```



Pasul 3

```
declare
    cod_client_max_pct_bonus number;
    v_clienti pachet_ex13.vector_clienti;
    v_coduri_filme pachet_ex13.tablou_imbricat;
begin
    -- PASUL 1: se adauga date in tabelele FILM, DIFUZEaza, REZERVARI
    -- + se activeaza triggerul pt rezervari (se modif implicit si pct bonus din tabela CLIENTI)
    --pachet_ex13.adauga_film_rezervare;

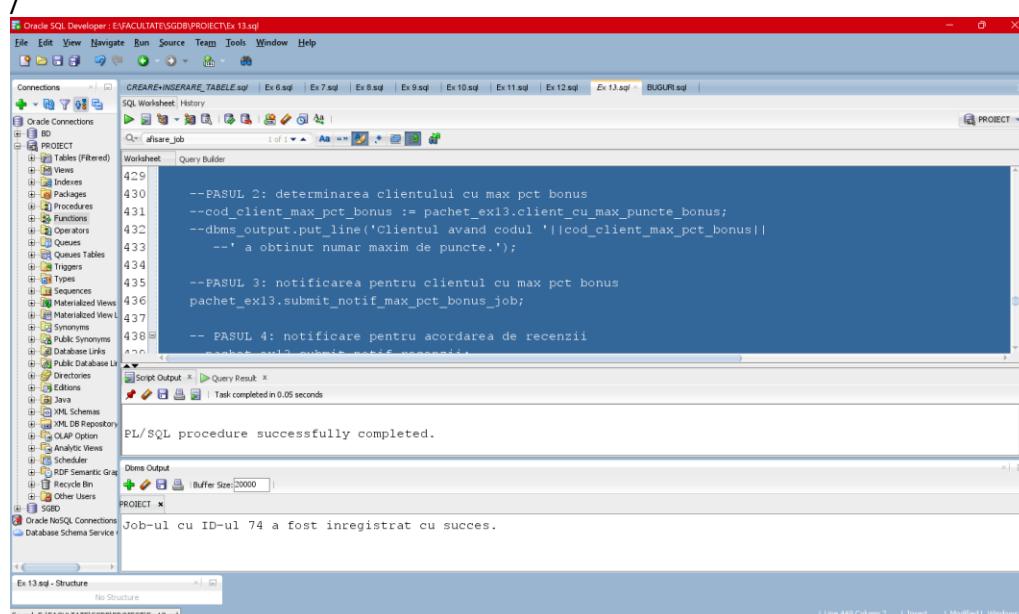
    --PASUL 2: determinarea clientului cu max pct bonus
    --cod_client_max_pct_bonus := pachet_ex13.client_cu_max_puncte_bonus;
    --dbms_output.put_line('Clientul avand codul '||cod_client_max_pct_bonus||
    --' a obtinut numar maxim de puncte.');

    --PASUL 3: notificarea pentru clientul cu max pct bonus
    pachet_ex13.submit_notif_max_pct_bonus_job;

    -- PASUL 4: notificare pentru acordarea de recenzii
    --pachet_ex13.submit_notif_recenzii;

    -- PASUL 5: adaugarea recenziilor
    --v_clienti := pachet_ex13.clienti_care_dau_recenzii;
    --pachet_ex13.adauga_recenzi(v_clienti);

    -- PASUL 6: afisarea clasamentului final al filmelor
    --v_coduri_filme := pachet_ex13.topul_filmelor_coduri;
    --pachet_ex13.afisare_top_filme(v_coduri_filme);
end;
/
```



```
PL/SQL procedure successfully completed.

Job-ul cu ID-ul 74 a fost inregistrat cu succes.
```

-- verificare PASUL 3

```
select job, next_date, what
from user_jobs;
```

The screenshot shows the Oracle SQL Developer interface. In the central workspace, a query builder window displays the following PL/SQL code:

```
-- verificare PASUL 3
select job, next_date, what
from user_jobs;

begin
    dbms_job.run(job => 74);
end;
```

Below the code, the results of the execution are shown in a table:

JOB	NEXT_DATE	WHAT
74	04-JAN-25 10:00:00	0000000000 PM +02:00 packet ex13.notif max pct bonus;

In the bottom right corner of the results pane, there is a message: "Job-ul cu ID-ul 74 a fost inregistrat cu succes."

The left sidebar shows various database objects like Tables, Views, Procedures, etc. A yellow arrow points to the "Editors" icon in the sidebar.

```
begin
```

```
    dbms_job.run(job => 74);
end;
/
```

The screenshot shows the Oracle SQL Developer interface again. The workspace contains the same PL/SQL code as the previous screenshot:

```
begin
    dbms_job.run(job => 74);
end;
/
```

The results pane shows the message: "PL/SQL procedure successfully completed." Below this, a notification message is displayed:

Notificare trimisa catre: Popescu Radu
Felicitari, ai castigat o vizionare gratuita din cabina de proiectie a oricarui film ales de tine!
Oferta este valabila pe perioada a doua saptamani incepand cu data la care ai primit mesajul.

The left sidebar shows the same list of database objects as the first screenshot.

Pasul 4

```
declare
    cod_client_max_pct_bonus number;
    v_clienti pachet_ex13.vector_clienti;
    v_coduri_filme pachet_ex13.tablou_imbricat;
begin
    -- PASUL 1: se adauga date in tabelele FILM, DIFUZEaza, REZERVARI
    -- + se activeaza triggerul pt rezervari (se modif implicit si pct bonus din tabela CLIENTI)
    --pachet_ex13.adauga_film_rezervare;

    --PASUL 2: determinarea clientului cu max pct bonus
    --cod_client_max_pct_bonus := pachet_ex13.client_cu_max_puncte_bonus;
    --dbms_output.put_line('Clientul avand codul '||cod_client_max_pct_bonus||
    --' a obtinut numar maxim de puncte.');

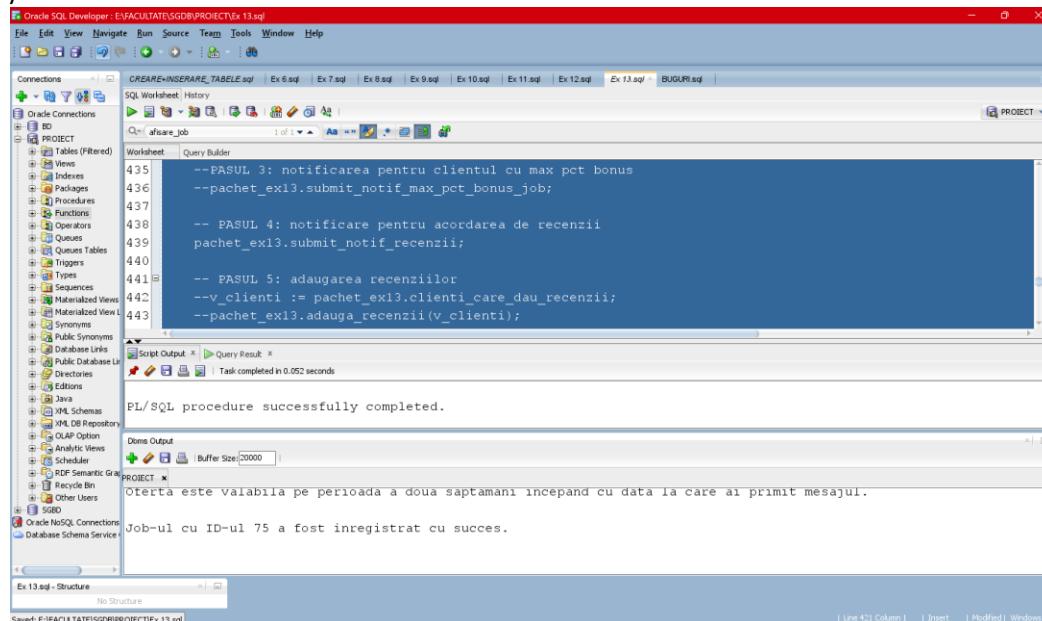
    --PASUL 3: notificarea pentru clientul cu max pct bonus
    --pachet_ex13.submit_notif_max_pct_bonus_job;

    -- PASUL 4: notificare pentru acordarea de recenzii
    pachet_ex13.submit_notif_recenzii;

    -- PASUL 5: adaugarea recenziilor
    --v_clienti := pachet_ex13.clienti_care_dau_recenzii;
    --pachet_ex13.adauga_recenzii(v_clienti);

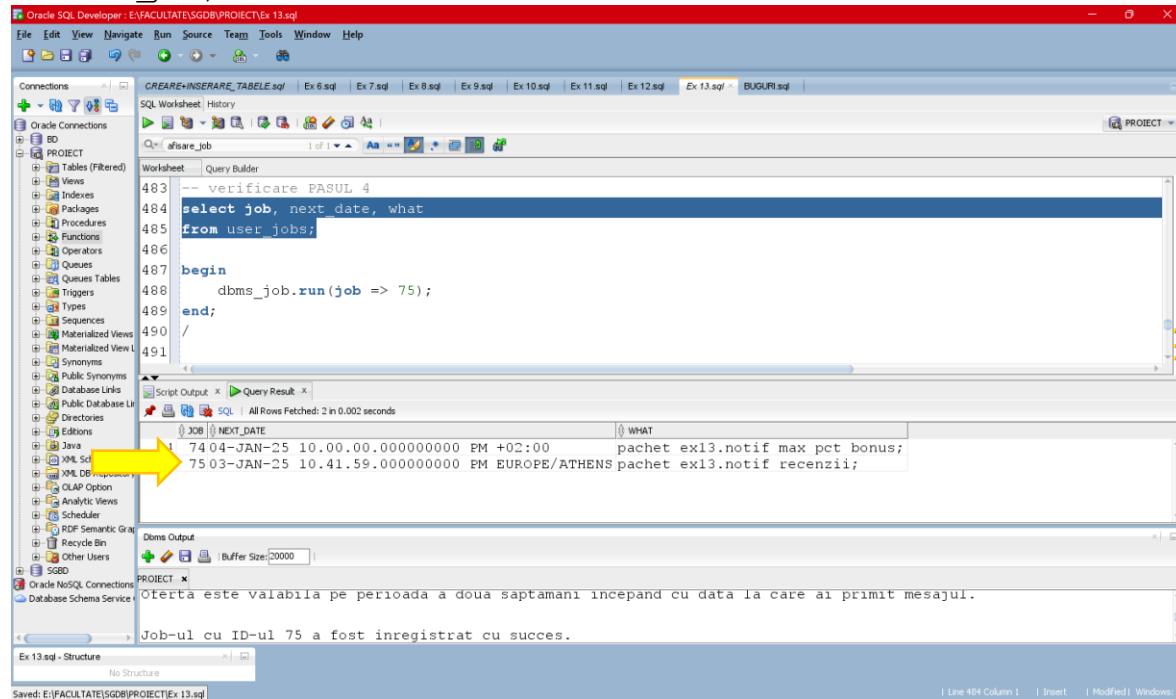
    -- PASUL 6: afisarea clasamentului final al filmelor
    --v_coduri_filme := pachet_ex13.topul_filmelor_coduri;
    --pachet_ex13.afisare_top_filme(v_coduri_filme);
end;
```

/



The screenshot shows the Oracle SQL Developer interface. The top menu bar includes File, Edit, View, Navigate, Run, Source, Team, Tools, Window, Help. The left sidebar shows the project structure with various database objects like Tables, Views, Indexes, Packages, Procedures, Functions, Operators, Queues, Triggers, Types, Sequences, Materialized View Log, Synonyms, Public Synonyms, Database Links, Public Database Links, Directories, Editions, PL/SQL Schemas, and Oracle NoSQL Connections. The main workspace has a 'Query Builder' window containing the PL/SQL code for step 4. Below it is a 'Script Output' window showing the message 'PL/SQL procedure successfully completed.' and a 'DML Output' window showing messages about offers and job registration. The bottom status bar indicates the file is saved at 'E:\FACULTATE\SGDB\PROJECT\Ex 13.sql'.

```
-- verificare PASUL 4
select job, next_date, what
from user_jobs;
```



Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex13.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections Oracle Connections BD PROJECT Tables (Filtered) Views Indexes Packages Procedures Functions Operators Queues Queues Tables Triggers Types Sequences Materialized Views Materialized View L Synonyms Public Synonyms Database Links Public Database Links Directories Editors Java XML Schema XML DB Repository OLAP Option Analytic Views Scheduler RDF Semantic Graph Recycle Bin Other Users SGBD Oracle NoSQL Connections Database Schema Service

CREATE+INSERARE_TABLEEx13.sql Ex 6.sql Ex 7.sql Ex 8.sql Ex 9.sql Ex 10.sql Ex 11.sql Ex 12.sql Ex 13.sql BUGURI.sql

SQL Worksheet History Q: afisare_job 1 of 1 Aa

Worksheet Query Builder

```
483 -- verificare PASUL 4
484 select job, next_date, what
485 from user_jobs;
486
487 begin
488   dbms_job.run(job => 75);
489 end;
490 /
491
```

Script Output Query Result

JOB	NEXT_DATE	WHAT
75	04-JAN-25 10.00.00.0000000 PM +02:00	pachet ex13.notif max pct bonus;
	03-JAN-25 10.41.59.0000000 PM EUROPE/ATHENS	pachet ex13.notif recenzie;

All Rows Fetched: 2 in 0.002 seconds

Outputs Output Buffer Size:20000

PROJECT

Oferta este valabila pe perioada a doua saptamani incepand cu data la care ai primit mesajul.

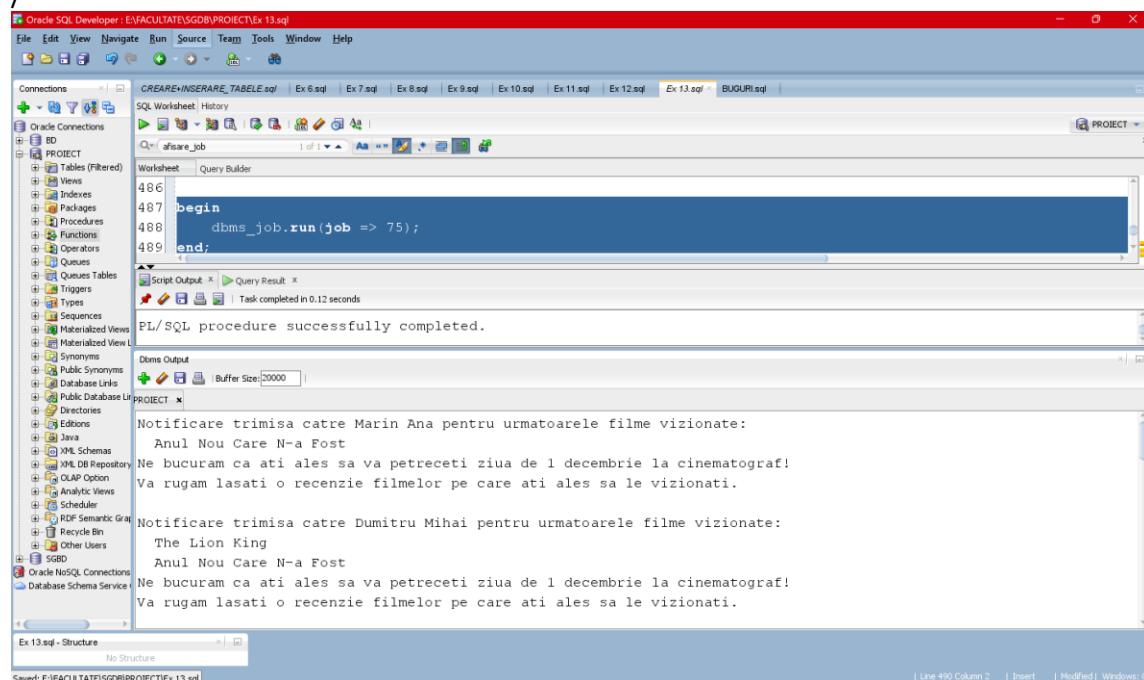
Job-ul cu ID-ul 75 a fost inregistrat cu succes.

Ex 13.sql - Structure No Structure

Saved: E:\FACULTATE\SGDB\PROJECT\Ex13.sql

Line 491 Column 1 Insert Modified Windows: C

```
begin
  dbms_job.run(job =>75);
end;
/
```



Oracle SQL Developer : E:\FACULTATE\SGDB\PROJECT\Ex13.sql

File Edit View Navigate Run Source Team Tools Window Help

Connections Oracle Connections BD PROJECT Tables (Filtered) Views Indexes Packages Procedures Functions Operators Queues Queues Tables Triggers Types Sequences Materialized Views Materialized View L Synonyms Public Synonyms Database Links Public Database Links Directories Editors Java XML Schema XML DB Repository OLAP Option Analytic Views Scheduler RDF Semantic Graph Recycle Bin Other Users SGBD Oracle NoSQL Connections Database Schema Service

CREATE+INSERARE_TABLEEx13.sql Ex 6.sql Ex 7.sql Ex 8.sql Ex 9.sql Ex 10.sql Ex 11.sql Ex 12.sql Ex 13.sql BUGURI.sql

SQL Worksheet History Q: afisare_job 1 of 1 Aa

Worksheet Query Builder

```
486
487 begin
488   dbms_job.run(job => 75);
489 end;
490 /
491
```

Script Output Query Result

PL/SQL procedure successfully completed.

Outputs Output Buffer Size:20000

PROJECT

Notificare trimisa catre Marin Ana pentru urmatoarele filme vizionate:
Anul Nou Care N-a Fost
Ne bucuram ca ati ales sa va petreceti ziua de 1 decembrie la cinematograf!
Va rugam lasati o recenzie filmelor pe care ati ales sa le vizionati.

Notificare trimisa catre Dumitru Mihai pentru urmatoarele filme vizionate:
The Lion King
Anul Nou Care N-a Fost
Ne bucuram ca ati ales sa va petreceti ziua de 1 decembrie la cinematograf!
Va rugam lasati o recenzie filmelor pe care ati ales sa le vizionati.

Ex 13.sql - Structure No Structure

Saved: E:\FACULTATE\SGDB\PROJECT\Ex13.sql

Line 490 Column 2 Insert Modified Windows: C

Pasul 5

```
declare
    cod_client_max_pct_bonus number;
    v_clienti pachet_ex13.vector_clienti;
    v_coduri_filme pachet_ex13.tablou_imbricat;
begin
    -- PASUL 1: se adauga date in tabelele FILM, DIFUZEaza, REZERVARI
    -- + se activeaza triggerul pt rezervari (se modif implicit si pct bonus din tabela CLIENTI)
    --pachet_ex13.adauga_film_rezervare;

    --PASUL 2: determinarea clientului cu max pct bonus
    --cod_client_max_pct_bonus := pachet_ex13.client_cu_max_puncte_bonus;
    --dbms_output.put_line('Clientul avand codul '||cod_client_max_pct_bonus||
    --' a obtinut numar maxim de puncte.');

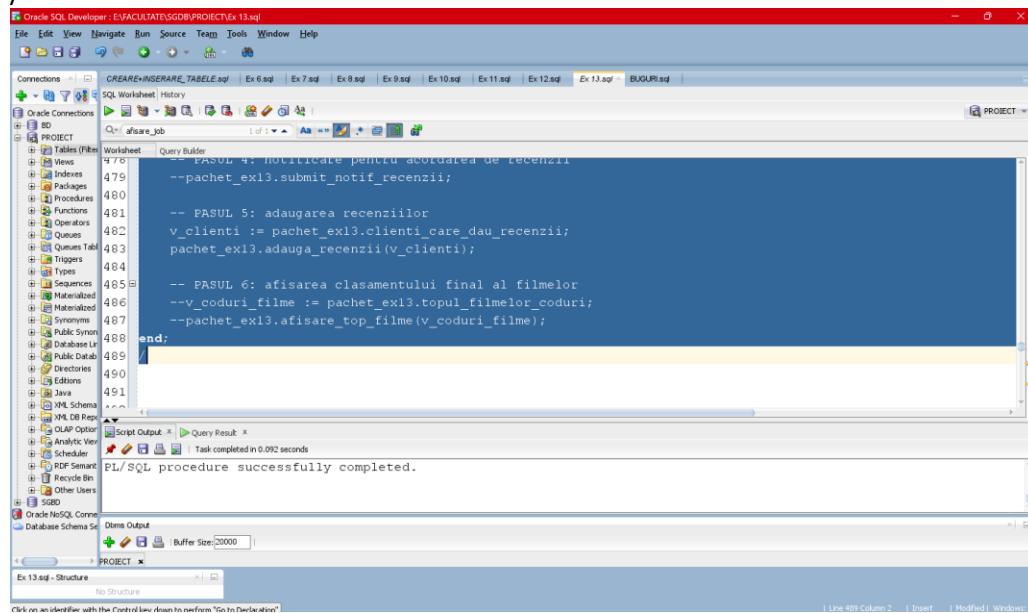
    --PASUL 3: notificarea pentru clientul cu max pct bonus
    --pachet_ex13.submit_notif_max_pct_bonus_job;

    -- PASUL 4: notificare pentru acordarea de recenzii
    --pachet_ex13.submit_notif_recenzii;

    -- PASUL 5: adaugarea recenziilor
    v_clienti := pachet_ex13.clienti_care_dau_recenzii;
    pachet_ex13.adauga_recenzi(v_clienti);

    -- PASUL 6: afisarea clasamentului final al filmelor
    --v_coduri_filme := pachet_ex13.topul_filmelor_coduri;
    --pachet_ex13.afisare_top_filme(v_coduri_filme);
end;
```

/



```
-- verificare PASUL 5
-- initial in tabelul RECENZII erau 31 de randuri, acum sunt 44
select * from recenzie;
```

COD_RECENZIE	COD_FILM	COD_CLIENT	SCOR	DATA_RECENZIE
29	R29	F17	10	10 13-NOV-24
30	R30	F19	10	6 16-DEC-24
31	R31	F23	1	7 05-DEC-24
32	R32	F28	4	7 02-DEC-24
33	R33	F17	5	3 02-DEC-24
34	R34	F28	5	7 02-DEC-24
35	R35	F27	7	2 02-DEC-24
36	R36	F17	8	8 02-DEC-24
37	R37	F17	10	3 02-DEC-24
38	R38	F26	10	9 02-DEC-24
39	R39	F26	11	1 02-DEC-24
40	R40	F26	12	8 02-DEC-24
41	R41	F17	17	9 02-DEC-24
42	R42	F28	17	8 02-DEC-24
43	R43	F28	21	7 02-DEC-24
44	R44	F27	22	2 02-DEC-24

Pasul 6

```
declare
    cod_client_max_pct_bonus number;
    v_clienti pachet_ex13.vector_clienti;
    v_coduri_filme pachet_ex13.tablou_imbricat;
begin
    -- PASUL 1: se adauga date in tabelele FILM, DIFUZEAZA, REZERVARI
    -- + se activeaza triggerul pt rezervari (se modif implicit si pct bonus din tabela CLIENTI)
    -- -pachet_ex13.adauga_film_rezervare;

    --PASUL 2: determinarea clientului cu max pct bonus
    --cod_client_max_pct_bonus := pachet_ex13.client_cu_max_puncte_bonus;
    --dbms_output.put_line('Clientul avand codul '||cod_client_max_pct_bonus|||
    --                      ' a obtinut numar maxim de puncte.');

    --PASUL 3: notificarea pentru clientul cu max pct bonus
    --pachet_ex13.submit_notif_max_pct_bonus_job;

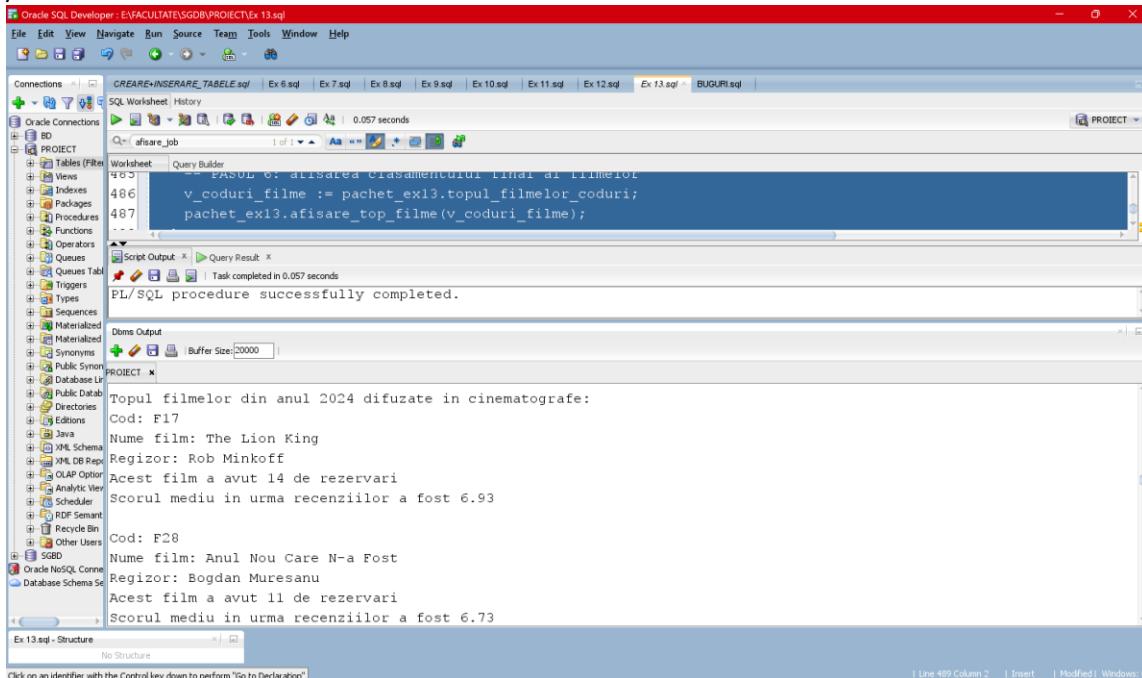
    -- PASUL 4: notificare pentru acordarea de recenzii
    --pachet_ex13.submit_notif_recenzii;
```

```
-- PASUL 5: adaugarea recenziiilor
--v_clienti := pachet_ex13.clienti_care_dau_recenzii;
--pachet_ex13.adauga_recenzii(v_clienti);
```

-- PASUL 6: afisarea clasamentului final al filmelor

```
v_coduri_filme := pachet_ex13.topul_filmelor_coduri;
pachet_ex13.afisare_top_filme(v_coduri_filme);
end;
```

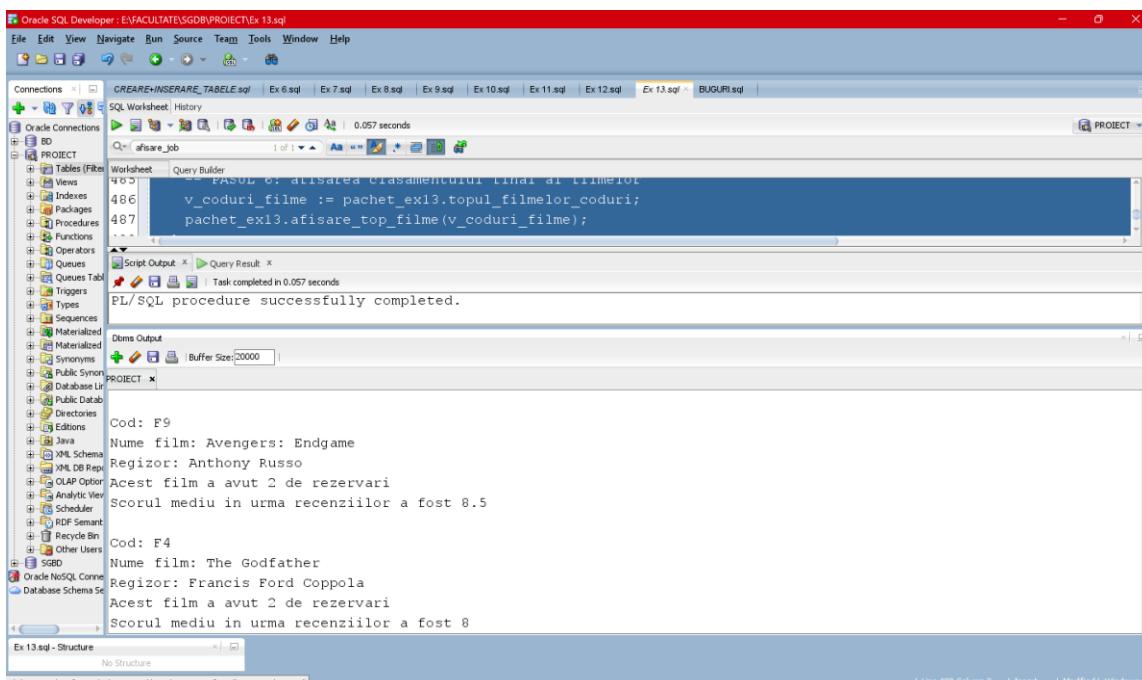
/



```
PL/SQL procedure successfully completed.

Dom Output
+ [ ] Buffer Size:20000 | PROJECT x
PROJECT x
Topul filmelor din anul 2024 difuzate in cinematografe:
Cod: F17
Nume film: The Lion King
Regizor: Rob Minkoff
Acest film a avut 14 de rezervari
Scorul mediu in urma recenziielor a fost 6.93

Cod: F28
Nume film: Anul Nou Care N-a Fost
Regizor: Bogdan Muresanu
Acest film a avut 11 de rezervari
Scorul mediu in urma recenziielor a fost 6.73
```



```
PL/SQL procedure successfully completed.

Dom Output
+ [ ] Buffer Size:20000 | PROJECT x
PROJECT x
Cod: F9
Nume film: Avengers: Endgame
Regizor: Anthony Russo
Acest film a avut 2 de rezervari
Scorul mediu in urma recenziielor a fost 8.5

Cod: F4
Nume film: The Godfather
Regizor: Francis Ford Coppola
Acest film a avut 2 de rezervari
Scorul mediu in urma recenziielor a fost 8
```