Proiect –Baze de Date

Gestionarea unui cinematograf

Bălăiță Cosmin – Neculai

Grupa 141

Cuprins

[1.Prezentare 3](#_Toc136020097)

[2. Constrângerile (restricții, reguli) impuse asupra modelului 4](#_Toc136020098)

[3. Descrierea entităților, incluzând precizarea cheii primare 5](#_Toc136020099)

[4.Descrierea relațiilor, incluzând precizarea cardinalității acestora. 5](#_Toc136020100)

[5. Descrierea atributelor 6](#_Toc136020101)

[6. ERD 8](#_Toc136020102)

[7. DIAGRAMA CONCEPTUALA 8](#_Toc136020103)

[8. Scheme corespunzatpare relatiilor 9](#_Toc136020104)

[9. Normalizarea FN1-FN3 9](#_Toc136020105)

[10. Secvența ce va fi utilizată în inserarea înregistrărilor în tabele 12](#_Toc136020106)

[11. Crearea tabelelor în SQL și inserarea de date coerente 12](#_Toc136020107)

[12. 5 cereri SQL complexe 23](#_Toc136020108)

[13. Implementarea a 3 operații de actualizare și de suprimare a datelor utilizând subcereri 26](#_Toc136020109)

[14. Crearea unei vizualizări complex 27](#_Toc136020110)

[15. Formulați în limbaj natural și implementați în SQL: o cerere ce utilizează operația outer-join pe minimum 4 tabele, o cerere ce utilizează operația division și o cerere care implementează analiza top-n.. 27](#_Toc136020111)

[16. Optimizarea unei cereri 28](#_Toc136020112)

# 1.Prezentare

Tema aleasa pentru proiect este Cinematograf.

O baza de date a unui cinematograf ar fi o structura de date care ar permite stocarea informațiilor despre filme, săli de cinema, programul de difuzare a filmelor si vânzările de bilete.

Modelul ar putea fi structurat astfel:

1. Film - Această entitate reprezintă un film care poate fi proiectat într-un cinematograf. Într-un model real, ar putea include informații despre titlu, descriere, durată, distribuție, buget, și încasări. Utilitatea acestei entități este de a permite unui cinematograf să stocheze și să acceseze informații despre filmele disponibile pentru proiecție, precum și să permită clienților să caute filme după diferite criterii, cum ar fi titlul sau genul.
2. Cinematograf - Această entitate reprezintă o locație fizică unde sunt proiectate filme. Într-un model real, ar putea include informații despre adresa, numărul de ecrane, programul de funcționare și numărul de locuri disponibile. Utilitatea acestei entități este de a permite unui cinematograf să stocheze informații despre locația și capacitatea sa, precum și să faciliteze procesul de achiziționare a biletelor și de planificare a proiecțiilor.
3. Sală - Această entitate reprezintă o sală specific într-un cinematograf unde se proiectează un film. Într-un model real ar include cinematograful din care face parte și ce film se difuzează în acea sală. Utilitatea acestei entități este de a permite unui cinematograf să stocheze informații despre sălile sale și să asocieze fiecare proiecție cu o sală specifică.
4. Ora - Această entitate reprezintă o oră specifică când un film este proiectatîntr-o anumită sală. Într-un model real, ar putea include informații despre data și ora de începere a proiecției și durata filmului. Utilitatea acestei entități este de a permite unui cinematograf să planifice proiecțiile și să vândă bilete pentru acestea.
5. Bilet - Această entitate reprezintă un bilet achiziționat pentru o oră de proiecție specifică. Într-un model real, ar putea include informații despre preț, locul și data de achiziționare, precum și orice oferte speciale sau reduceri aplicate. Utilitatea acestei entități este de a permite unui cinematograf să urmărească vânzările de bilete și să gestioneze locurile disponibile într-o sală de proiecție.
6. Scaun - Această entitate reprezintă un scaun într-o sală specific. Într-un model real, ar putea include informații despre locația sa în sala de proiecție și alte detalii despre confortul scaunelor. Utilitatea acestei entități este de a permite unui cinematograf să gestioneze locurile disponibile într-o sală de proiecție și să asocieze fiecare scaun cu un bilet specific.
7. Client- utilitatea acesteia ar fi de a permite unui cinematograf să urmărească informații despre clienți și să gestioneze vânzările de bilete. Aceasta ar putea include informații despre numele, adresa și informațiile de contact ale clientului, precum și informații despre biletele achiziționate.
8. Angajat - utilitatea acesteia ar fi de a permite unui cinematograf să urmărească informații despre angajați și să gestioneze programul de lucru și sarcinile alocate acestora. Aceasta ar putea include informații despre numele, adresa și informațiile de contact ale angajatului, precum și informații despre poziția sa în cadrul companiei și programul de lucru.
9. Gen - utilitatea acesteia ar fi de a permite unui cinematograf să clasifice filmele în diferite categorii și să permită clienților să caute filme după gen. Aceasta ar putea include informații despre numele și descrierea genului.
10. Rating - utilitatea acesteia ar fi de a permite unui cinematograf să clasifice filmele în funcție de ratingul lor și să permită clienților să caute filme în funcție de acest rating. Aceasta ar putea include informații despre numele și descrierea ratingului.
11. Regizor - utilitatea acesteia ar fi de a permite unui cinematograf să urmărească informații despre regizorii filmelor și să permită clienților să caute filme după regizor. Aceasta ar putea include informații despre numele, biografia și filmografia regizorului.
12. Actor - utilitatea acesteia ar fi de a permite unui cinematograf să urmărească informații despre actorii dintr-un film și să permită clienților să caute filme după actori. Aceasta ar putea include informații despre numele, biografia și filmografia actorului.

Regulile de funcționare pentru baza de date ar trebui sa permită adăugarea, stocarea si accesarea datelor, precum si efectuarea de interogări pentru a genera rapoarte si statistici relevante. Utilitățile acestei baze de date ar include:

* gestionarea programului de difuzare a filmelor;
* gestionarea vânzărilor de bilete;
* gestionarea dotărilor si capacitații sălilor de cinema;
* generarea de rapoarte si statistici pentru performanta afacerii si a vânzărilor de bilete.

In final, aceasta baza de date ar permite companiei sa gestioneze eficient interacțiunea cu clienții si sa ofere o experiența personalizata, dar si sa ia decizii informate legate de programul de difuzare a filmelor si alocarea resurselor.

# 2. Constrângerile (restricții, reguli) impuse asupra modelului

1. Restrângerea de unicat - Unele tabele pot fi restrânse la înregistrări unice, cum ar fi tabela "Sali", care ar trebui să conțină doar înregistrări unice pentru fiecare sală disponibilă.
2. Chei străine - Tabelele vor fi legate între ele prin chei străine pentru a permite extragerea de informații în mod eficient și coerent.
3. Validarea datelor - Se vor impune reguli de validare a datelor pentru a asigura că datele introduse în baza de date sunt valide și complete, cum ar fi verificarea validității adresei de e-mail a clienților sau a datelor și orarului spectacolelor.
4. Reguli de afaceri - Se vor impune reguli de afaceri specifice pentru a asigura că procesul de achiziție și rezervare se desfășoară într-un mod eficient și coerent, cum ar fi numărul maxim de bilete pe care un client le poate achiziționa sau numărul maxim de zile înainte de spectacol în care poate face o rezervare.
5. Validarea relațiilor - Validarea relațiilor dintre tabele este importantă pentru a preveni erori de referință sau inconsistente, de exemplu, asigurându-se că id-urile spectacolelor și biletele sunt legate numai la înregistrări existente în tabelele corespunzătoare.
6. Limitări - Limitările pot fi impuse pentru a asigura că utilizatorii nu introduc date care depășesc anumite valori maxime, cum ar fi numărul maxim de bilete pe care un client le poate achiziționa sau suma maximă pe care o pot cheltui.

# 3. Descrierea entităților, incluzând precizarea cheii primare

1. Film - Reprezintă un film care este proiectat în cinematograf. Cheia primară ar fi ID-ul filmului (ID\_film).

2. Cinematograf - Reprezintă o locație fizică unde sunt proiectate filme. Cheia primară ar fi ID-ul cinematografului (ID\_cinematograf).

3. Ecran - Reprezintă un ecran specific în cadrul unui cinematograf unde se proiectează un film. Cheia primară ar fi ID-ul ecranului (ID\_ecran).

4. Ora de proiecție - Reprezintă o oră specifică când un film este proiectat pe un anumit ecran. Cheia primară ar fi ID-ul orei de proiecție (ID\_ora\_proiectie).

5. Bilet - Reprezintă un bilet achiziționat pentru o oră de proiecție specifică. Cheia primară ar fi ID-ul biletului (ID\_bilet).

6. Scaun - Reprezintă un scaun într-un ecran specific. Cheia primară ar fi ID-ul scaunului (ID\_scaun).

7. Client - Reprezintă o persoană care achiziționează un bilet. Cheia primară ar fi ID-ul clientului (ID\_client).

8. Angajat - Reprezintă un membru al personalului care lucrează la cinematograf. Cheia primară ar fi ID-ul angajatului (ID\_angajat).

9. Gen - Reprezintă o categorie de filme, cum ar fi acțiune, dramă sau comedie. Cheia primară ar fi ID-ul genului (ID\_gen).

10. Rating - Reprezintă un rating pentru un film, cum ar fi G, PG, PG-13 sau R. Cheia primară ar fi ID-ul ratingului (ID\_rating).

11. Regizor - Reprezintă regizorul unui film. Cheia primară ar fi ID-ul regizorului (ID\_regizor).

12. Actor - Reprezintă un actor care lucreaza pentru un regizor. Cheia primară ar fi ID-ul actorului (ID\_actor).4.

# 4.Descrierea relațiilor, incluzând precizarea cardinalității acestora.

* Film - Regizor: relație de tip many - to - one(un film poate fi regizat de un singur regizor, dar un regizor poate regiza mai multe filme)
* Film - Gen: relație de tip many - to - many (un film poate avea mai multe genuri, iar un gen poate fi asociat cu mai multe filme)
* Film - Rating: relație de tip many - to - one(un film poate avea un singur rating, dar un rating poate fi asociat cu mai multe filme)
* Regizor - Actor: relație de tip one - to - many (un regizor poate avea mai mulți actori, iar un actor poate fi asociat unui singur regizor)
* Ecran - Film: relație de tip many - to - one(un ecran poate fi asociat cu un singur film, dar un film poate fi proiectat pe mai multe ecrane)
* Ecran - Scaun: relație de tip many - to - many (un ecran poate avea mai multe scaune, iar un scaun poate fi asociat cu mai multe ecrane)
* Ecran - Program: relație de tip many - to - manymulți (un ecran poate avea mai multe programe de difuzare, dar un program poate fi asociat cu un singur ecran)
* Program - Film: relație de tip many - to - one(un program poate fi asociat cu un singur film, dar un film poate fi programat în mai multe programe)
* Bilet - Client: relație de tip many - to - one(un bilet poate fi achiziționat de un singur client, dar un client poate achiziționa mai multe bilete)
* Bilet - Scaun: relație de tip many - to - one(un bilet poate fi asociat cu un singur scaun, dar un scaun poate fi asociat cu mai multe bilete)
* Angajat - Program: relație de tip many - to - many (un angajat poate fi alocat mai multor programe, iar un program poate avea mai mulți angajați alocati)
* Angajat - Sarcina: relație de tip many - to - many (un angajat poate avea mai multe sarcini, iar o sarcina poate fi asignată la mai mulți angajați)

# 5. Descrierea atributelor

1. Film:

* ID\_FILM (primary key)-int
* TITLU (varchar2)
* AN\_APARITIE (int)
* DURATA (int)
* ID\_GEN (foreign key)
* ID\_RATING (foreign key)
* ID\_REGIZOR (foreign key)

1. Cinematograf:

* ID\_CINEMATOGRAF (primary key)-int
* NUME (varchar2)
* ORAS (varchar2)
* NUMAR\_SALI

1. Ecran:

* ID\_ECRAN (primary key)-int
* ID\_CINEMATOGRAF (foreign key)
* ID\_SCAUN (foreign key)

1. Ora\_de\_proiectie

* ID\_ORA (primary key)-int
* ORA\_INCEPUT (time)
* ORA\_SFARSIT (time)
* DATA (date)
* ID\_ECRAN (foreign key)
* ID\_FILM(foreign key)

1. Bilet:

* ID\_BILET (primary key)-int
* ID\_CLIENT (foreign key)
* ID\_SCAUN (foreign key)
* ID\_ORA (foreign key)
* PRET (int)

1. Scaun:

* ID\_SCAUN (primary key)-int
* ID\_ECRAN (foreign key)
* NUMAR (int)
* RAND (int)

1. Client:

* ID\_CLIENT (primary key)-int
* NUME (varchar2)
* PRENUME (varchar2)
* DATA\_NASTERII (date)
* TELEFON (varchar2)
* EMAIL(varchar2)

1. Angajat:

* ID\_ANGAJAT (primary key)-int
* NUME (varchar2)
* PRENUME (varchar2)
* DATA\_NASTERII (date)
* SALARIU (int)
* FUNCTIE (varchar2)
* ID\_CINEMATOGRAF (foreign key)

1. Gen:

* ID\_GEN (primary key)-int
* DENUMIRE (varchar2)

1. Rating:

* ID\_RATING (primary key)-int
* valoare (double)

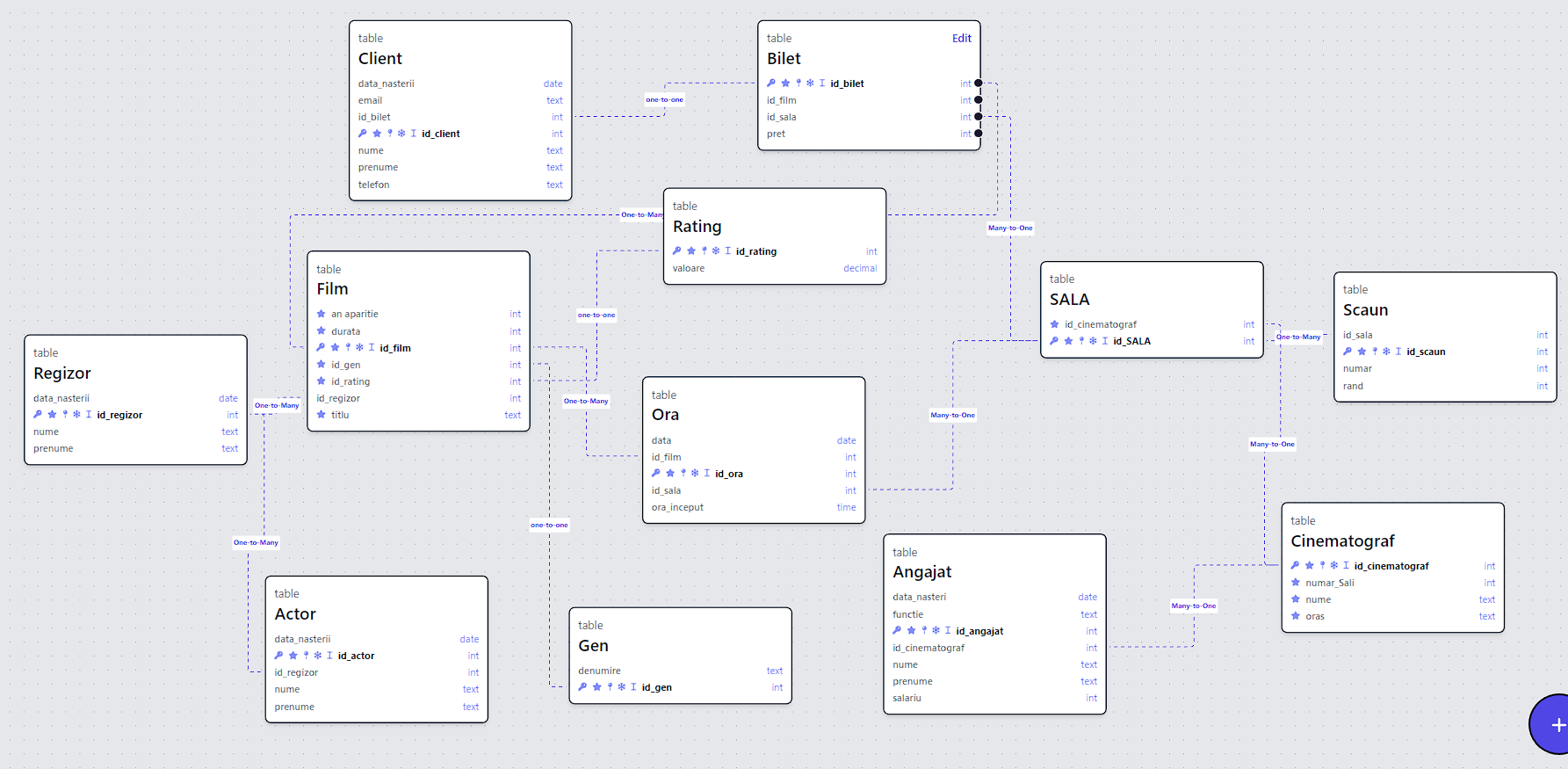
1. Regizor:

* ID\_REGIZOR (primary key)-int
* NUME (varchar2)
* PRENUME(varchar2)
* DATA\_NASTERII (date)
* ID\_FILM (foreign key)

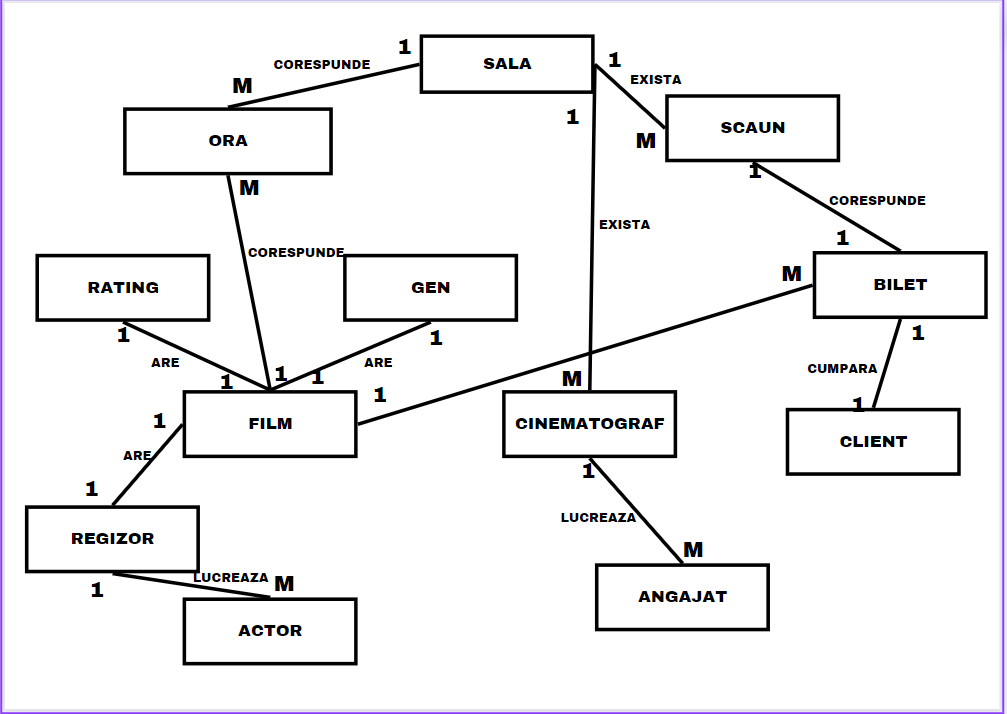
1. Actor:

* ID\_ACTOR(primary key)-int
* NUME (varchar2)
* PRENUME (varchar2)
* DATA\_NASTERII (date)
* ID\_REGIZOR (foreign key)

# 6. ERD



# 7. DIAGRAMA CONCEPTUALA



# 8. Scheme corespunzatpare relatiilor

Cinematograf(ID\_cinematograf, Nume, Adresa, Numar\_sali)

Sala(ID\_Sala, ID\_cinematograf, ID\_film)

Ora (ID\_ora, Ora\_inceput, Data, ID\_SALA, ID\_film)

Film(id\_fim, titlu, an\_aparitie, durata, id\_gen, id\_rating, id\_regizor)

Bilet(ID\_bilet, ID\_client, ID\_film, ID\_sala, Pret)

Scaun(ID\_scaun, ID\_SALA,Numar, Rand)

Client(ID\_client, Nume, Prenume, Data\_nasterii, Adresa, Email, Telefon)

Angajat(ID\_angajat, Nume, Prenume, Data\_nasterii, Salariu, Functie, ID\_cinematograf)

Gen(ID\_gen, Denumire)

Rating(ID\_rating, Denumire)

Regizor(ID\_regizor, Nume, Prenume, Data\_nastere)

Actor(ID\_actor, Nume, Prenume, Data\_nastere)

# 9. Normalizarea FN1-FN3

**Tabelele în from inițială:**

Tabelul "Cinematograf"

Cinematograf(ID\_cinematograf, Nume, Adresa, Numar\_sali, Nume\_film, Ora\_inceput, Data\_film)

Tabelul "Sala"

Sala(ID\_Sala, ID\_cinematograf, ID\_film, Nume\_cinematograf, Adresa\_cinematograf, Numar\_sali, Nume\_film, Ora\_inceput, Data\_film)

Tabelul "Ora"

Ora(ID\_ora, Ora\_inceput, Data, ID\_Sala, ID\_cinematograf, Nume\_cinematograf, Adresa\_cinematograf, Numar\_sali, Nume\_film)

Tabelul "Film"

Film(id\_film, titlu, an\_aparitie, durata, id\_gen, id\_rating, id\_regizor, Nume\_gen, Denumire\_rating, Nume\_regizor)

Tabelul "Bilet"

Bilet(ID\_bilet, ID\_client, ID\_film, ID\_sala, Pret, Nume\_client, Prenume\_client, Nume\_film, Nume\_cinematograf, Adresa\_cinematograf, Numar\_sali)

Tabelul "Scaun"

Scaun(ID\_scaun, ID\_SALA,Numar, Rand, ID\_cinematograf, Nume\_cinematograf, Adresa\_cinematograf, Numar\_sali, Nume\_film, Ora\_inceput, Data\_film)

Tabelul "Client"

Client(ID\_client, Nume, Prenume, Data\_nasterii, Adresa, Email, Telefon)

Tabelul "Angajat"

Angajat(ID\_angajat, Nume, Prenume, Data\_nasterii, Salariu, Functie, ID\_cinematograf, Nume\_cinematograf, Adresa\_cinematograf, Numar\_sali)

Tabelul "Gen"

Gen(ID\_gen, Denumire)

Tabelul "Rating"

Rating(ID\_rating, Denumire)

Tabelul "Regizor"

Regizor(ID\_regizor, Nume, Prenume, Data\_nastere)

Tabelul "Actor"

Actor(ID\_actor, Nume, Prenume, Data\_nastere)

**Pentru a realiza normalizarea până la Forma Normală de Nivelul 1 (FN1), Forma Normală de Nivelul 2 (FN2) și Forma Normală de Nivelul 3 (FN3), vom analiza fiecare tabel în parte și vom aplica regulile de normalizare.**

**Tabelul "Cinematograf" (ID\_cinematograf, Nume, Adresa, Numar\_sali):**

- Respectă FN1 deoarece fiecare atribut conține o singură valoare atomică.

- Respectă FN2 deoarece nu există dependențe parțiale sau tranzitive.

- Respectă FN3 deoarece nu există dependențe tranzitive.

**Tabelul "Sala" (ID\_Sala, ID\_cinematograf, ID\_film):**

- Respectă FN1 deoarece fiecare atribut conține o singură valoare atomică.

- Respectă FN2 deoarece nu există dependențe parțiale sau tranzitive.

- Respectă FN3 deoarece nu există dependențe tranzitive.

**Tabelul "Ora" (ID\_ora, Ora\_inceput, Data, ID\_SALA, ID\_film):**

- Respectă FN1 deoarece fiecare atribut conține o singură valoare atomică.

- Respectă FN2 deoarece nu există dependențe parțiale sau tranzitive.

- Respectă FN3 deoarece nu există dependențe tranzitive.

**Tabelul "Film" (id\_film, titlu, an\_aparitie, durata, id\_gen, id\_rating, id\_regizor):**

- Respectă FN1 deoarece fiecare atribut conține o singură valoare atomică.

- Respectă FN2 deoarece nu există dependențe parțiale sau tranzitive.

- Respectă FN3 deoarece nu există dependențe tranzitive.

**Tabelul "Bilet" (ID\_bilet, ID\_client, ID\_film, ID\_sala, Pret):**

- Respectă FN1 deoarece fiecare atribut conține o singură valoare atomică.

- Respectă FN2 deoarece nu există dependențe parțiale sau tranzitive.

- Respectă FN3 deoarece nu există dependențe tranzitive.

**Tabelul "Scaun" (ID\_scaun, ID\_SALA, Numar, Rand):**

- Respectă FN1 deoarece fiecare atribut conține o singură valoare atomică.

- Respectă FN2 deoarece nu există dependențe parțiale sau tranzitive.

- Respectă FN3 deoarece nu există dependențe tranzitive.

**Tabelul "Client" (ID\_client, Nume, Prenume, Data\_nasterii, Adresa, Email, Telefon):**

- Respectă FN1 deoarece fiecare atribut conține o singură valoare atomică.

- Respectă FN2 deoarece nu există dependențe parțiale sau tranzitive.

- Respectă FN3 deoarece nu există dependențe tranzitive.

**Tabelul "Angajat" (ID\_angajat, Nume, Prenume, Data\_nasterii, Salariu, Functie, ID\_cinematograf):**

- Respectă FN1 deoarece fiecare atribut conține o singură valoare atomică.

- Respectă FN2 deoarece nu există dependențe parțiale sau tranzitive.

- Respectă FN3 deoarece nu există dependențe tranzitive.

**Tabelul "Gen" (ID\_gen, Denumire):**

- Respectă FN1 deoarece fiecare atribut conține o singură valoare atomică.

- Respectă FN2 deoarece nu există dependențe parțiale sau tranzitive.

- Respectă FN3 deoarece nu există dependențe tranzitive.

**Tabelul "Rating" (ID\_rating, Denumire):**

- Respectă FN1 deoarece fiecare atribut conține o singură valoare atomică.

- Respectă FN2 deoarece nu există dependențe parțiale sau tranzitive.

- Respectă FN3 deoarece nu există dependențe tranzitive.

**Tabelul "Regizor" (ID\_regizor, Nume, Prenume, Data\_nastere):**

- Respectă FN1 deoarece fiecare atribut conține o singură valoare atomică.

- Respectă FN2 deoarece nu există dependențe parțiale sau tranzitive.

- Respectă FN3 deoarece nu există dependențe tranzitive.

**Tabelul "Actor" (ID\_actor, Nume, Prenume, Data\_nastere):**

- Respectă FN1 deoarece fiecare atribut conține o singură valoare atomică.

- Respectă FN2 deoarece nu există dependențe parțiale sau tranzitive.

- Respectă FN3 deoarece nu există dependențe tranzitive.

Toate tabelele se află deja în Forma Normală de Nivelul 1 (FN1), Forma Normală de Nivelul 2 (FN2) și Forma Normală de Nivelul 3 (FN3), deci respectă toate regulile de normalizare.

# 10. Secvența ce va fi utilizată în inserarea înregistrărilor în tabele

CREATE SEQUENCE CINEMATOGRAF\_SEQ

START WITH 1

INCREMENT BY 1

NOCACHE

NOCYCLE;

INSERT INTO CINEMATOGRAF (ID\_CINEMATOGRAF, NUME, ORAS, NUMAR\_SALI)

VALUES (CINEMATOGRAF\_SEQ.NEXTVAL, 'CINEMA MAGIC', 'BACAU', 1);

# 11. Crearea tabelelor în SQL și inserarea de date coerente

**CREARE TABELE:**

---CREARE TABEL CINEMATOGRAF

CREATE TABLE CINEMATOGRAF

(

ID\_CINEMATOGRAF INT PRIMARY KEY,

NUME VARCHAR(100) NOT NULL,

ORAS VARCHAR(200) NOT NULL,

NUMAR\_SALI INT

);

---CREARE TABEL GEN

CREATE TABLE GEN

(

ID\_GEN INT PRIMARY KEY,

DENUMIRE VARCHAR2(100)

);

---creare tabel bilet

CREATE TABLE BILET

(

ID\_BILET INT PRIMARY KEY,

PRET INT,

ID\_FILM INT,

ID\_SALA INT,

FOREIGN KEY (ID\_FILM) REFERENCES FILM (ID\_FILM),

FOREIGN KEY (ID\_SALA) REFERENCES SALA (ID\_SALA)

);

---CREARE TABEL CLIENTI

CREATE TABLE CLIENTI

(

ID\_CLIENT INT PRIMARY KEY,

NUME VARCHAR2(100),

PRENUME VARCHAR2(100),

TELEFON VARCHAR2(100) UNIQUE,

EMAIL VARCHAR2(200) UNIQUE,

DATA\_NASTERII DATE,

ID\_BILET INT,

FOREIGN KEY (ID\_BILET) REFERENCES BILET (ID\_BILET)

);

ALTER TABLE CLIENTI

ADD CONSTRAINT CHK\_CLIENTI\_EMAIL

CHECK (REGEXP\_LIKE(EMAIL, '^[a-zA-Z0-9.\_%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$'));

---CREARE TABEL RATING

CREATE TABLE RATING

(

ID\_RATING INT PRIMARY KEY,

VALOARE INT

);

---CREARE TABEL REGIZOR

CREATE TABLE REGIZOR

(

ID\_REGIZOR INT PRIMARY KEY,

NUME VARCHAR2(100),

PRENUME VARCHAR2(100),

DATA\_NASTERII DATE

);

---CREARE TABELA ACTOR

CREATE TABLE ACTOR

(

ID\_ACTOR INT PRIMARY KEY,

NUME VARCHAR2(100),

PRENUME VARCHAR2(100),

DATA\_NASTERII DATE,

ID\_REGIZOR INT,

FOREIGN KEY (ID\_REGIZOR) REFERENCES REGIZOR (ID\_REGIZOR)

);

---CREARE TABEL FILM

CREATE TABLE FILM

(

ID\_FILM INT PRIMARY KEY,

TITLU VARCHAR2(200),

AN\_APARITIE INT,

DURATA INT,

ID\_REGIZOR INT,

ID\_GEN INT,

ID\_RATING INT,

FOREIGN KEY (ID\_REGIZOR) REFERENCES REGIZOR (ID\_REGIZOR),

FOREIGN KEY (ID\_GEN) REFERENCES GEN (ID\_GEN),

FOREIGN KEY (ID\_RATING) REFERENCES RATING (ID\_RATING)

);

---CREARE TABEL SALA

CREATE TABLE SALA

(

ID\_SALA INT PRIMARY KEY,

ID\_CINEMATOGRAF INT,

FOREIGN KEY (ID\_CINEMATOGRAF) REFERENCES CINEMATOGRAF (ID\_CINEMATOGRAF)

);

---CREARE TABEL SCAUN

CREATE TABLE SCAUN

(

ID\_SCAUN INT PRIMARY KEY,

NUMAR INT,

RAND INT,

ID\_SALA INT,

FOREIGN KEY (ID\_SALA) REFERENCES SALA (ID\_SALA)

);

--- CREARE TABELA ANGAJAT

CREATE TABLE ANGAJAT

(

ID\_ANGAJAT INT PRIMARY KEY,

NUME VARCHAR2 (100),

PRENUME VARCHAR2 (100),

DATA\_NASTERII DATE,

FUNCTIE VARCHAR2 (50),

SALARIU INT,

ID\_CINEMATOGRAF INT,

FOREIGN KEY (ID\_CINEMATOGRAF)REFERENCES CINEMATOGRAF (ID\_CINEMATOGRAF)

);

---creare tabel ora de proiectare

CREATE TABLE ORA (

ID\_ORA INT PRIMARY KEY,

DATA\_FILM DATE,

ORA TIMESTAMP,

ID\_FILM INT,

ID\_SALA INT,

FOREIGN KEY (ID\_FILM) REFERENCES FILM (ID\_FILM),

FOREIGN KEY (ID\_SALA) REFERENCES SALA (ID\_SALA)

);

**Inserare valori in tabele**

---INSERARE IN TABELUL CINEMATOGRAF

INSERT INTO CINEMATOGRAF (ID\_CINEMATOGRAF, NUME, ORAS, NUMAR\_SALI)

VALUES (1, 'CINEMA CITY', 'CLUJ', 5);

INSERT INTO CINEMATOGRAF (ID\_CINEMATOGRAF, NUME, ORAS, NUMAR\_SALI)

VALUES (2, 'CINEMA MALL', 'IASI', 3);

INSERT INTO CINEMATOGRAF (ID\_CINEMATOGRAF, NUME, ORAS, NUMAR\_SALI)

VALUES (3, 'CINEMA BANEASA', 'BUCURESTI', 10);

INSERT INTO CINEMATOGRAF (ID\_CINEMATOGRAF, NUME, ORAS, NUMAR\_SALI)

VALUES (4, 'CINEMA BACAU', 'BACAU', 2);

INSERT INTO CINEMATOGRAF (ID\_CINEMATOGRAF, NUME, ORAS, NUMAR\_SALI)

VALUES (5, 'CINEMA MAGIC', 'BACAU', 1);

---INSERARE VALORI IN TABELUL GEN

INSERT INTO GEN (ID\_GEN, DENUMIRE)

VALUES (1, 'DRAMA');

INSERT INTO GEN (ID\_GEN, DENUMIRE)

VALUES (2, 'SCI-FI');

INSERT INTO GEN (ID\_GEN, DENUMIRE)

VALUES (3, 'ROMANTIC');

INSERT INTO GEN (ID\_GEN, DENUMIRE)

VALUES (4, 'HORROR');

INSERT INTO GEN (ID\_GEN, DENUMIRE)

VALUES (5, 'COMEDIE');

INSERT INTO GEN (ID\_GEN, DENUMIRE)

VALUES (6, 'ANIMATIE');

INSERT INTO GEN (ID\_GEN, DENUMIRE)

VALUES (7, 'DOCUMENTAR');

INSERT INTO GEN (ID\_GEN, DENUMIRE)

VALUES (8, 'THRILLER');

---INSRARE VALORI IN TABELUL RATING

INSERT INTO RATING (ID\_RATING, VALOARE)

VALUES (1, 1);

INSERT INTO RATING (ID\_RATING, VALOARE)

VALUES (2, 2);

INSERT INTO RATING (ID\_RATING, VALOARE)

VALUES (3, 3);

INSERT INTO RATING (ID\_RATING, VALOARE)

VALUES (4, 4);

INSERT INTO RATING (ID\_RATING, VALOARE)

VALUES (5, 5);

---INSERARE IN TABELUL REGIZOR

INSERT INTO REGIZOR (ID\_REGIZOR, NUME, PRENUME, DATA\_NASTERII)

VALUES (1, 'SMITH', 'JOHN', TO\_DATE('1990-05-15', 'YYYY-MM-DD'));

INSERT INTO REGIZOR (ID\_REGIZOR, NUME, PRENUME, DATA\_NASTERII)

VALUES (2, 'JOHNSON', 'EMILY', TO\_DATE('1985-10-20', 'YYYY-MM-DD'));

INSERT INTO REGIZOR (ID\_REGIZOR, NUME, PRENUME, DATA\_NASTERII)

VALUES (3, 'WILLIAMS', 'MICHAEL', TO\_DATE('1978-03-12', 'YYYY-MM-DD'));

INSERT INTO REGIZOR (ID\_REGIZOR, NUME, PRENUME, DATA\_NASTERII)

VALUES (4, 'BROWN', 'SOPHIA', TO\_DATE('1992-08-02', 'YYYY-MM-DD'));

INSERT INTO REGIZOR (ID\_REGIZOR, NUME, PRENUME, DATA\_NASTERII)

VALUES (5, 'DAVIS', 'DANIEL', TO\_DATE('1987-11-25', 'YYYY-MM-DD'));

INSERT INTO REGIZOR (ID\_REGIZOR, NUME, PRENUME, DATA\_NASTERII)

VALUES (6, 'MILLER', 'OLIVIA', TO\_DATE('1994-02-18', 'YYYY-MM-DD'));

INSERT INTO REGIZOR (ID\_REGIZOR, NUME, PRENUME, DATA\_NASTERII)

VALUES (7, 'WILSON', 'ANDREW', TO\_DATE('1980-09-05', 'YYYY-MM-DD'));

INSERT INTO REGIZOR (ID\_REGIZOR, NUME, PRENUME, DATA\_NASTERII)

VALUES (8, 'MOORE', 'ISABELLA', TO\_DATE('1983-06-08', 'YYYY-MM-DD'));

INSERT INTO REGIZOR (ID\_REGIZOR, NUME, PRENUME, DATA\_NASTERII)

VALUES (9, 'TAYLOR', 'DAVID', TO\_DATE('1998-12-30', 'YYYY-MM-DD'));

INSERT INTO REGIZOR (ID\_REGIZOR, NUME, PRENUME, DATA\_NASTERII)

VALUES (10, 'ANDERSON', 'SOPHIE', TO\_DATE('1991-04-10', 'YYYY-MM-DD'));

---INSEARRE IN TABELUL FILM

INSERT INTO FILM (ID\_FILM, TITLU, AN\_APARITIE, DURATA, ID\_REGIZOR, ID\_GEN, ID\_RATING)

VALUES (1, 'TITANIC', 1997, 194, 1, 3, 4);

INSERT INTO FILM (ID\_FILM, TITLU, AN\_APARITIE, DURATA, ID\_REGIZOR, ID\_GEN, ID\_RATING)

VALUES (11, 'AVATAR', 2009, 195,1, 3,4);

INSERT INTO FILM (ID\_FILM, TITLU, AN\_APARITIE, DURATA, ID\_REGIZOR, ID\_GEN, ID\_RATING)

VALUES (2, 'INCEPTION', 2010, 148, 2, 1, 5);

INSERT INTO FILM (ID\_FILM, TITLU, AN\_APARITIE, DURATA, ID\_REGIZOR, ID\_GEN, ID\_RATING)

VALUES (3, 'THE SHAWSHANK REDEMPTION', 1994, 142, 3, 2, 5);

INSERT INTO FILM (ID\_FILM, TITLU, AN\_APARITIE, DURATA, ID\_REGIZOR, ID\_GEN, ID\_RATING)

VALUES (4, 'PULP FICTION', 1994, 154, 4, 4, 4);

INSERT INTO FILM (ID\_FILM, TITLU, AN\_APARITIE, DURATA, ID\_REGIZOR, ID\_GEN, ID\_RATING)

VALUES (5, 'THE DARK KNIGHT', 2008, 152, 5, 1, 5);

INSERT INTO FILM (ID\_FILM, TITLU, AN\_APARITIE, DURATA, ID\_REGIZOR, ID\_GEN, ID\_RATING)

VALUES (6, 'THE GODFATHER', 1972, 175, 6, 2, 5);

INSERT INTO FILM (ID\_FILM, TITLU, AN\_APARITIE, DURATA, ID\_REGIZOR, ID\_GEN, ID\_RATING)

VALUES (7, 'FIGHT CLUB', 1999, 139, 7, 3, 4);

INSERT INTO FILM (ID\_FILM, TITLU, AN\_APARITIE, DURATA, ID\_REGIZOR, ID\_GEN, ID\_RATING)

VALUES (8, 'FORREST GUMP', 1994, 142, 8, 2, 5);

INSERT INTO FILM (ID\_FILM, TITLU, AN\_APARITIE, DURATA, ID\_REGIZOR, ID\_GEN, ID\_RATING)

VALUES (9, 'THE MATRIX', 1999, 136, 9, 1, 4);

INSERT INTO FILM (ID\_FILM, TITLU, AN\_APARITIE, DURATA, ID\_REGIZOR, ID\_GEN, ID\_RATING)

VALUES (10, 'GOODFELLAS', 1990, 146, 10, 4, 4);

---INSEARE IN TABELUL SALA

INSERT INTO SALA (ID\_SALA, ID\_CINEMATOGRAF)

VALUES (1, 1);

INSERT INTO SALA (ID\_SALA, ID\_CINEMATOGRAF)

VALUES (2, 1);

INSERT INTO SALA (ID\_SALA, ID\_CINEMATOGRAF)

VALUES (3, 1);

INSERT INTO SALA (ID\_SALA, ID\_CINEMATOGRAF)

VALUES (4, 1);

INSERT INTO SALA (ID\_SALA, ID\_CINEMATOGRAF)

VALUES (5, 1);

INSERT INTO SALA (ID\_SALA, ID\_CINEMATOGRAF)

VALUES (6, 2);

INSERT INTO SALA (ID\_SALA, ID\_CINEMATOGRAF)

VALUES (7, 2);

INSERT INTO SALA (ID\_SALA, ID\_CINEMATOGRAF)

VALUES (8, 2);

INSERT INTO SALA (ID\_SALA, ID\_CINEMATOGRAF)

VALUES (9, 3);

INSERT INTO SALA (ID\_SALA, ID\_CINEMATOGRAF)

VALUES (10, 3);

INSERT INTO SALA (ID\_SALA, ID\_CINEMATOGRAF)

VALUES (11, 3);

INSERT INTO SALA (ID\_SALA, ID\_CINEMATOGRAF)

VALUES (12, 3);

INSERT INTO SALA (ID\_SALA, ID\_CINEMATOGRAF)

VALUES (13, 3);

INSERT INTO SALA (ID\_SALA, ID\_CINEMATOGRAF)

VALUES (14, 3);

INSERT INTO SALA (ID\_SALA, ID\_CINEMATOGRAF)

VALUES (15, 4);

INSERT INTO SALA (ID\_SALA, ID\_CINEMATOGRAF)

VALUES (16, 5);

---INSERARE IN TABELA SCAUN

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (1, 1, 1, 1);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (2, 2, 1, 1);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (3, 3, 2, 1);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (4, 4, 2, 1);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (5, 1, 1, 2);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (6, 2, 1, 2);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (7, 3, 2, 2);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (8, 4, 2, 2);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (9, 1, 1, 3);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (10, 2, 1, 3);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (11, 3, 2, 3);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (12, 4, 2, 3);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (13, 1, 1, 4);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (14, 2, 1, 4);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (15, 3, 2, 4);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (16, 4, 2, 4);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (17, 1, 1, 5);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (18, 2, 1, 5);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (19, 3, 2, 5);

INSERT INTO SCAUN (ID\_SCAUN, NUMAR, RAND, ID\_SALA)

VALUES (20, 4, 2, 4);

---INSERARE VALORI IN TABELA ACTOR

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (1, 'DE NIRO', 'ROBERT', TO\_DATE('1943-08-17', 'YYYY-MM-DD'), 1);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (2, 'PITT', 'BRAD', TO\_DATE('1963-12-18', 'YYYY-MM-DD'), 2);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (3, 'WINSLET', 'KATE', TO\_DATE('1975-10-05', 'YYYY-MM-DD'), 3);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (4, 'DICAPRIO', 'LEONARDO', TO\_DATE('1974-11-11', 'YYYY-MM-DD'), 4);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (5, 'STREEP', 'MERYL', TO\_DATE('1949-06-22', 'YYYY-MM-DD'), 5);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (6, 'CRUISE', 'TOM', TO\_DATE('1962-07-03', 'YYYY-MM-DD'), 6);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (7, 'LAWRENCE', 'JENNIFER', TO\_DATE('1990-08-15', 'YYYY-MM-DD'), 7);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (8, 'HANKS', 'TOM', TO\_DATE('1956-07-09', 'YYYY-MM-DD'), 8);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (9, 'ROBERTS', 'JULIA', TO\_DATE('1967-10-28', 'YYYY-MM-DD'), 9);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (10, 'WASHINGTON', 'DENZEL', TO\_DATE('1954-12-28', 'YYYY-MM-DD'), 10);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (11, 'BULLOCK', 'SANDRA', TO\_DATE('1964-07-26', 'YYYY-MM-DD'), 1);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (12, 'HATHAWAY', 'ANNE', TO\_DATE('1982-11-12', 'YYYY-MM-DD'), 2);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (13, 'CUMBERBATCH', 'BENEDICT', TO\_DATE('1976-07-19', 'YYYY-MM-DD'), 3);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (14, 'JOHANSSON', 'SCARLETT', TO\_DATE('1984-11-22', 'YYYY-MM-DD'), 4);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (15, 'SMITH', 'WILL', TO\_DATE('1968-09-25', 'YYYY-MM-DD'), 5);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (16, 'PORTMAN', 'NATALIE', TO\_DATE('1981-06-09', 'YYYY-MM-DD'), 6);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (17, 'GOSLING', 'RYAN', TO\_DATE('1980-11-12', 'YYYY-MM-DD'), 7);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (18, 'BLUNT', 'EMILY', TO\_DATE('1983-02-23', 'YYYY-MM-DD'), 8);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (19, 'DAMON', 'MATT', TO\_DATE('1970-10-08', 'YYYY-MM-DD'), 9);

INSERT INTO ACTOR (ID\_ACTOR, NUME, PRENUME, DATA\_NASTERII, ID\_REGIZOR)

VALUES (20, 'KIDMAN', 'NICOLE', TO\_DATE('1967-06-20', 'YYYY-MM-DD'), 10);

---INSERARE VALORI IN TABELA ANGAJAT

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (1, 'Popescu', 'Ion', TO\_DATE('1990-05-15', 'YYYY-MM-DD'), 'Manager', 5000, 1);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (2, 'Ionescu', 'Maria', TO\_DATE('1995-08-20', 'YYYY-MM-DD'), 'Casier', 3000, 1);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (3, 'Constantinescu', 'Alexandru', TO\_DATE('1992-03-12', 'YYYY-MM-DD'), 'Agent de securitate', 3500, 1);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (4, 'Dumitru', 'Andreea', TO\_DATE('1993-11-25', 'YYYY-MM-DD'), 'Bucătar', 2800, 1);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (5, 'Stanescu', 'George', TO\_DATE('1988-07-05', 'YYYY-MM-DD'), 'Manager', 4800, 2);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (6, 'Mihai', 'Ana', TO\_DATE('1994-02-18', 'YYYY-MM-DD'), 'Casier', 3200, 2);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (7, 'Petrescu', 'Cristian', TO\_DATE('1991-09-30', 'YYYY-MM-DD'), 'Agent de securitate', 3500, 2);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (8, 'Popa', 'Andrei', TO\_DATE('1996-12-12', 'YYYY-MM-DD'), 'Bucătar', 2800, 2);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (9, 'Gheorghe', 'Ioana', TO\_DATE('1993-03-17', 'YYYY-MM-DD'), 'Manager', 5000, 3);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (10, 'Munteanu', 'Adrian', TO\_DATE('1989-06-28', 'YYYY-MM-DD'), 'Casier', 3000, 3);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (11, 'Voicu', 'Roxana', TO\_DATE('1994-09-02', 'YYYY-MM-DD'), 'Agent de securitate', 3500, 3);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (12, 'Popescu', 'Andrei', TO\_DATE('1997-11-10', 'YYYY-MM-DD'), 'Bucătar', 2800, 3);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (14, 'Iacob', 'Simona', TO\_DATE('1990-08-15', 'YYYY-MM-DD'), 'Manager', 4800, 4);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (15, 'Dragomir', 'Mihai', TO\_DATE('1996-01-20', 'YYYY-MM-DD'), 'Casier', 3200, 4);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (16, 'Radu', 'Cristina', TO\_DATE('1993-06-12', 'YYYY-MM-DD'), 'Agent de securitate', 3500, 4);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (17, 'Dinu', 'Gabriel', TO\_DATE('1997-09-25', 'YYYY-MM-DD'), 'Bucătar', 2800, 4);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (18, 'Popa', 'Alexandra', TO\_DATE('1989-07-05', 'YYYY-MM-DD'), 'Manager', 5000, 5);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (19, 'Ionescu', 'Adrian', TO\_DATE('1995-02-18', 'YYYY-MM-DD'), 'Casier', 3000, 5);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (20, 'Stan', 'Marian', TO\_DATE('1992-09-30', 'YYYY-MM-DD'), 'Agent de securitate', 3500, 5);

INSERT INTO Angajat (ID\_ANGAJAT, NUME, PRENUME, DATA\_NASTERII, FUNCTIE, SALARIU, ID\_CINEMATOGRAF)

VALUES (21, 'Popescu', 'Maria', TO\_DATE('1996-12-12', 'YYYY-MM-DD'), 'Bucătar', 2800, 5);

---inseare valori in tabele ora de proiectare

INSERT INTO ORA (ID\_ORA, DATA\_FILM, ORA, ID\_FILM, ID\_SALA)

VALUES(1, TO\_DATE('2023-05-16', 'YYYY-MM-DD'), TO\_TIMESTAMP('12:00:00', 'HH24:MI:SS'), 1, 1);

INSERT INTO ORA (ID\_ORA, DATA\_FILM, ORA, ID\_FILM, ID\_SALA)

VALUES(2, TO\_DATE('2023-05-16', 'YYYY-MM-DD'), TO\_TIMESTAMP('14:30:00', 'HH24:MI:SS'), 2, 2);

INSERT INTO ORA (ID\_ORA, DATA\_FILM, ORA, ID\_FILM, ID\_SALA)

VALUES(3, TO\_DATE('2023-05-16', 'YYYY-MM-DD'), TO\_TIMESTAMP('17:00:00', 'HH24:MI:SS'), 3, 3);

INSERT INTO ORA (ID\_ORA, DATA\_FILM, ORA, ID\_FILM, ID\_SALA)

VALUES(4, TO\_DATE('2023-05-16', 'YYYY-MM-DD'), TO\_TIMESTAMP('19:30:00', 'HH24:MI:SS'), 4, 4);

INSERT INTO ORA (ID\_ORA, DATA\_FILM, ORA, ID\_FILM, ID\_SALA)

VALUES(5, TO\_DATE('2023-05-16', 'YYYY-MM-DD'), TO\_TIMESTAMP('21:00:00', 'HH24:MI:SS'), 5, 5);

INSERT INTO ORA (ID\_ORA, DATA\_FILM, ORA, ID\_FILM, ID\_SALA)

VALUES(6, TO\_DATE('2023-05-16', 'YYYY-MM-DD'), TO\_TIMESTAMP('12:30:00', 'HH24:MI:SS'), 6, 1);

INSERT INTO ORA (ID\_ORA, DATA\_FILM, ORA, ID\_FILM, ID\_SALA)

VALUES(7, TO\_DATE('2023-05-16', 'YYYY-MM-DD'), TO\_TIMESTAMP('15:00:00', 'HH24:MI:SS'), 7, 2);

INSERT INTO ORA (ID\_ORA, DATA\_FILM, ORA, ID\_FILM, ID\_SALA)

VALUES(8, TO\_DATE('2023-05-16', 'YYYY-MM-DD'), TO\_TIMESTAMP('17:30:00', 'HH24:MI:SS'), 8, 3);

INSERT INTO ORA (ID\_ORA, DATA\_FILM, ORA, ID\_FILM, ID\_SALA)

VALUES(9, TO\_DATE('2023-05-16', 'YYYY-MM-DD'), TO\_TIMESTAMP('20:00:00', 'HH24:MI:SS'), 1, 4);

INSERT INTO ORA (ID\_ORA, DATA\_FILM, ORA, ID\_FILM, ID\_SALA)

VALUES(10, TO\_DATE('2023-05-16', 'YYYY-MM-DD'), TO\_TIMESTAMP('22:30:00', 'HH24:MI:SS'), 2, 5);

INSERT INTO ORA (ID\_ORA, DATA\_FILM, ORA, ID\_FILM, ID\_SALA)

VALUES(11, TO\_DATE('2023-05-16', 'YYYY-MM-DD'), TO\_TIMESTAMP('13:00:00', 'HH24:MI:SS'), 3, 6);

INSERT INTO ORA (ID\_ORA, DATA\_FILM, ORA, ID\_FILM, ID\_SALA)

VALUES(12, TO\_DATE('2023-05-16', 'YYYY-MM-DD'), TO\_TIMESTAMP('15:30:00', 'HH24:MI:SS'), 4, 7);

INSERT INTO ORA (ID\_ORA, DATA\_FILM, ORA, ID\_FILM, ID\_SALA)

VALUES(13, TO\_DATE('2023-05-16', 'YYYY-MM-DD'), TO\_TIMESTAMP('18:00:00', 'HH24:MI:SS'), 5, 8);

INSERT INTO ORA (ID\_ORA, DATA\_FILM, ORA, ID\_FILM, ID\_SALA)

VALUES(14, TO\_DATE('2023-05-16', 'YYYY-MM-DD'), TO\_TIMESTAMP('20:30:00', 'HH24:MI:SS'), 6, 9);

INSERT INTO ORA (ID\_ORA, DATA\_FILM, ORA, ID\_FILM, ID\_SALA)

VALUES(15, TO\_DATE('2023-05-16', 'YYYY-MM-DD'), TO\_TIMESTAMP('22:45:00', 'HH24:MI:SS'), 7, 10);

INSERT INTO ORA (ID\_ORA, DATA\_FILM, ORA, ID\_FILM, ID\_SALA)

VALUES(16, TO\_DATE('2023-05-16', 'YYYY-MM-DD'), TO\_TIMESTAMP('13:30:00', 'HH24:MI:SS'), 8, 11);

---inseare valori in tabela bilet

INSERT INTO Bilet (ID\_Bilet, ID\_FILM, ID\_SALA, Pret)

VALUES (1, 1, 1, 25.00);

INSERT INTO Bilet (ID\_Bilet, ID\_FILM, ID\_SALA, Pret)

VALUES (2, 2, 2, 25.50);

INSERT INTO Bilet (ID\_Bilet, ID\_FILM, ID\_SALA, Pret)

VALUES (3, 3, 3, 30.00);

INSERT INTO Bilet (ID\_Bilet, ID\_FILM, ID\_SALA, Pret)

VALUES (4, 4, 4, 30.00);

INSERT INTO Bilet (ID\_Bilet, ID\_FILM, ID\_SALA, Pret)

VALUES (5, 5, 5, 20.00);

INSERT INTO Bilet (ID\_Bilet, ID\_FILM, ID\_SALA, Pret)

VALUES (6, 6, 6, 35.00);

INSERT INTO Bilet (ID\_Bilet, ID\_FILM, ID\_SALA, Pret)

VALUES (7, 7, 7, 35.00);

INSERT INTO Bilet (ID\_Bilet, ID\_FILM, ID\_SALA, Pret)

VALUES (8, 8, 8, 18.00);

INSERT INTO Bilet (ID\_Bilet, ID\_FILM, ID\_SALA, Pret)

VALUES (9, 9, 9, 18.00);

INSERT INTO Bilet (ID\_Bilet, ID\_FILM, ID\_SALA, Pret)

VALUES (10, 10, 10, 22.50);

SELECT \* FROM BILET;

---inseare valori in tabela clienti

INSERT INTO CLIENTI (ID\_CLIENT, Nume, Prenume, Data\_Nasterii, Email, Telefon, id\_bilet)

VALUES (1, 'Popescu', 'Ion', TO\_DATE('1990-01-01','YYYY-MM-DD'), 'ion.popescu@gmail.com', '0123456789',1);

INSERT INTO CLIENTI (ID\_CLIENT, Nume, Prenume, Data\_Nasterii, Email, Telefon, id\_bilet)

VALUES (2, 'Ionescu', 'Maria',TO\_DATE('1985-05-15','YYYY-MM-DD') , 'maria.ionescu@gmail.com', '9876543210',2);

INSERT INTO CLIENTI (ID\_CLIENT, Nume, Prenume, Data\_Nasterii, Email, Telefon, id\_bilet)

VALUES (3, 'Popa', 'Ana', TO\_DATE('1992-12-10','YYYY-MM-DD'), 'ana.popa@gmail.com', '5555555555',3);

INSERT INTO CLIENTI (ID\_CLIENT, Nume, Prenume, Data\_Nasterii, Email, Telefon, id\_bilet)

VALUES (4, 'Georgescu', 'Mihai',TO\_DATE('1988-07-20','YYYY-MM-DD') , 'mihai.georgescu@gmail.com', '1234567890',4);

INSERT INTO CLIENTI (ID\_CLIENT, Nume, Prenume, Data\_Nasterii, Email, Telefon, id\_bilet)

VALUES (5, 'Radu', 'Elena', TO\_DATE('1994-04-05','YYYY-MM-DD'), 'elena.radu@gmail.com', '9999999999',5);

INSERT INTO CLIENTI (ID\_CLIENT, Nume, Prenume, Data\_Nasterii, Email, Telefon, id\_bilet)

VALUES (6, 'Stefanescu', 'Andrei',TO\_DATE('1991-09-18','YYYY-MM-DD') , 'andrei.stefanescu@gmail.com', '7777777777',6);

INSERT INTO CLIENTI (ID\_CLIENT, Nume, Prenume, Data\_Nasterii, Email, Telefon, id\_bilet)

VALUES (7, 'Dumitru', 'Cristina', TO\_DATE('1987-02-28','YYYY-MM-DD'), 'cristina.dumitru@gmail.com', '5556555555',7);

INSERT INTO CLIENTI (ID\_CLIENT, Nume, Prenume, Data\_Nasterii, Email, Telefon, id\_bilet)

VALUES (8, 'Constantin', 'Alexandru', TO\_DATE('1993-11-12','YYYY-MM-DD'), 'alexandru.constantin@gmail.com', '4444444444',8);

INSERT INTO CLIENTI(ID\_CLIENT, Nume, Prenume, Data\_Nasterii, Email, Telefon, id\_bilet)

VALUES (9, 'Mihai', 'Andreea', TO\_DATE('1989-06-08','YYYY-MM-DD'), 'andreea.mihai@gmail.com', '2222222222',9);

INSERT INTO CLIENTI (ID\_CLIENT, Nume, Prenume, Data\_Nasterii, Email, Telefon, id\_bilet)

VALUES (10, 'Gheorghe', 'Vlad', TO\_DATE('1996-03-25','YYYY-MM-DD'), 'vlad.gheorghe@gmail.com', '8888888888',10);

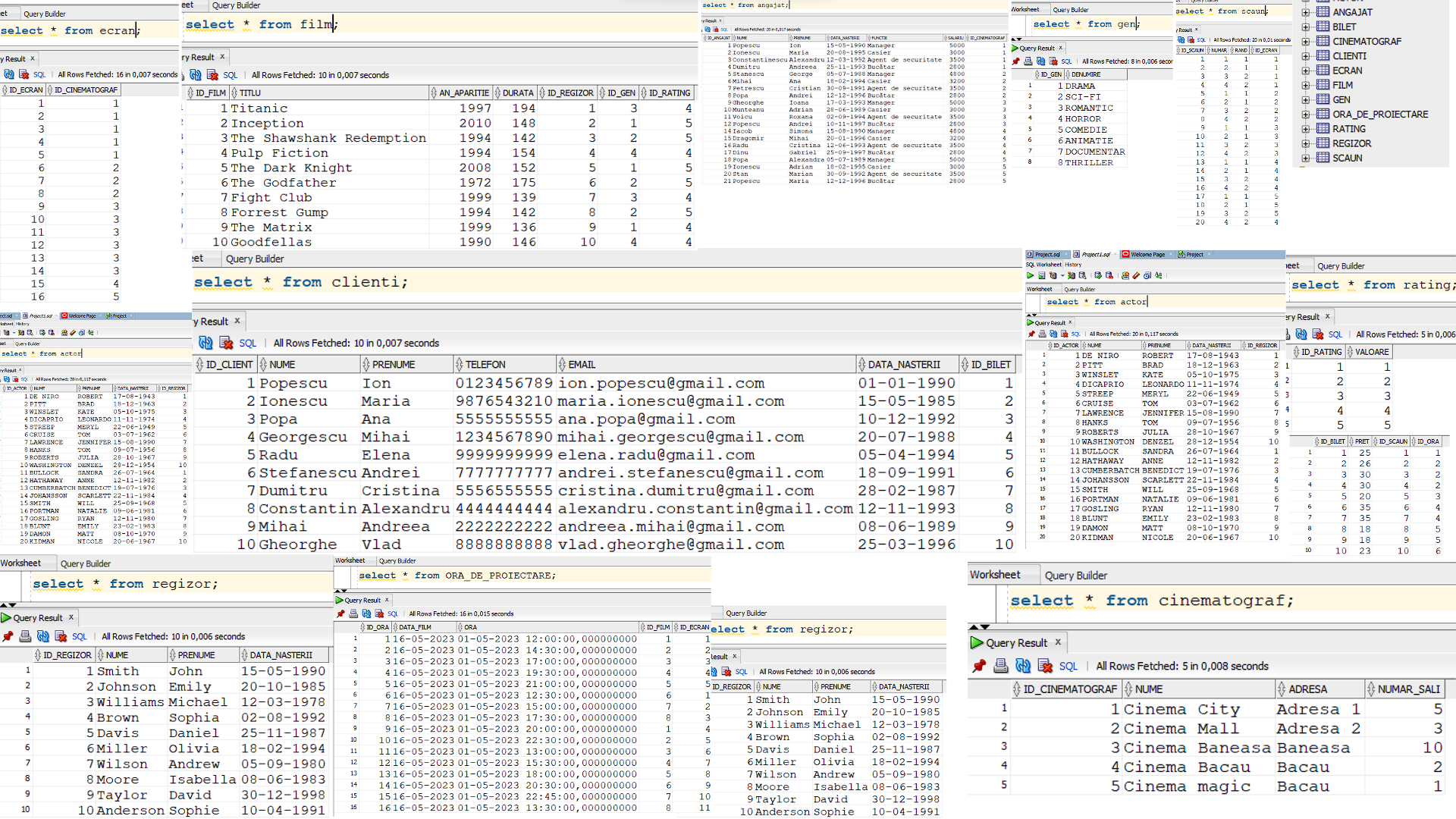
INSERT INTO CLIENTI (ID\_CLIENT, NUME)

VALUES (11, 'Vladimir');

INSERT INTO BILET(ID\_BILET)

VALUES(11);

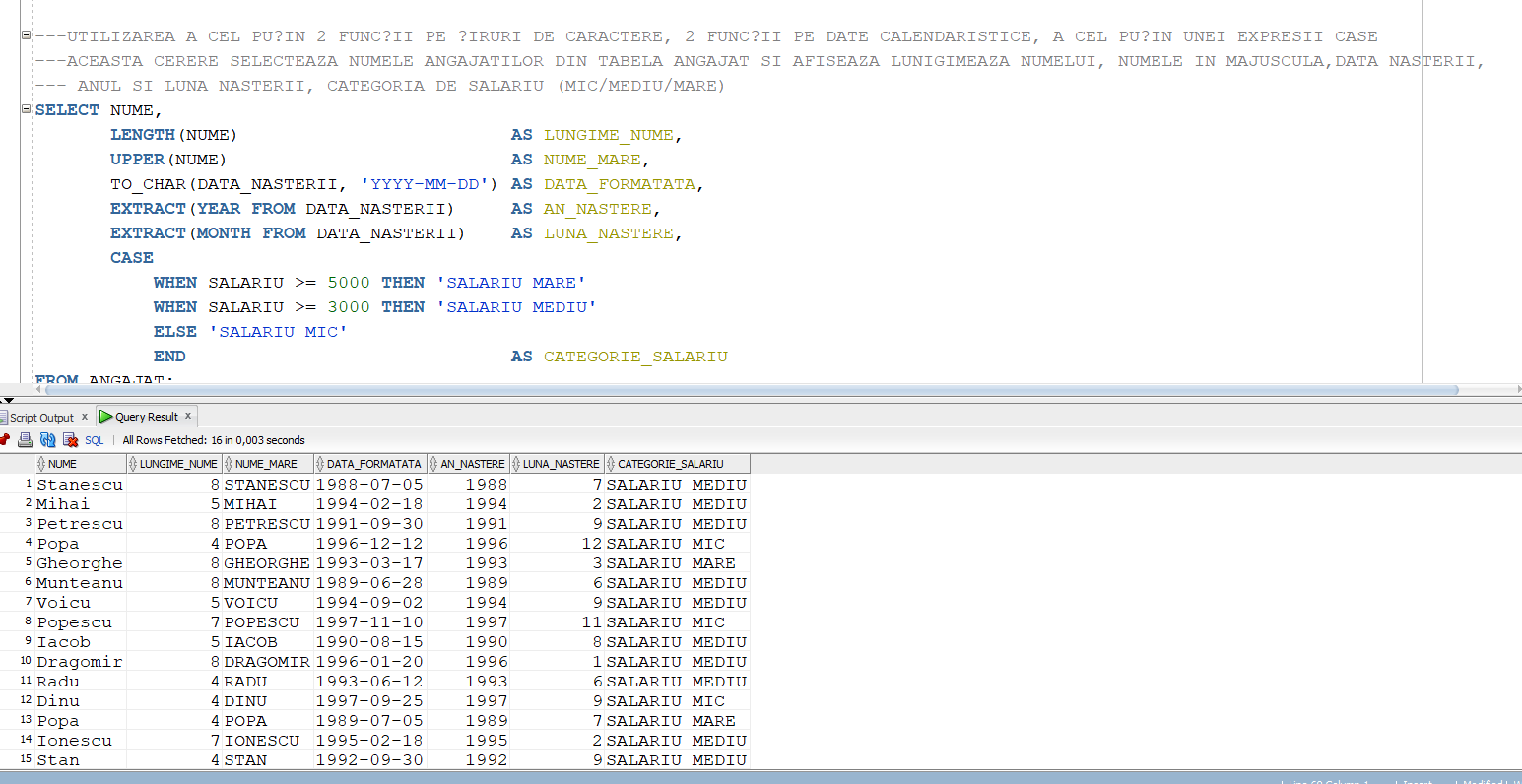
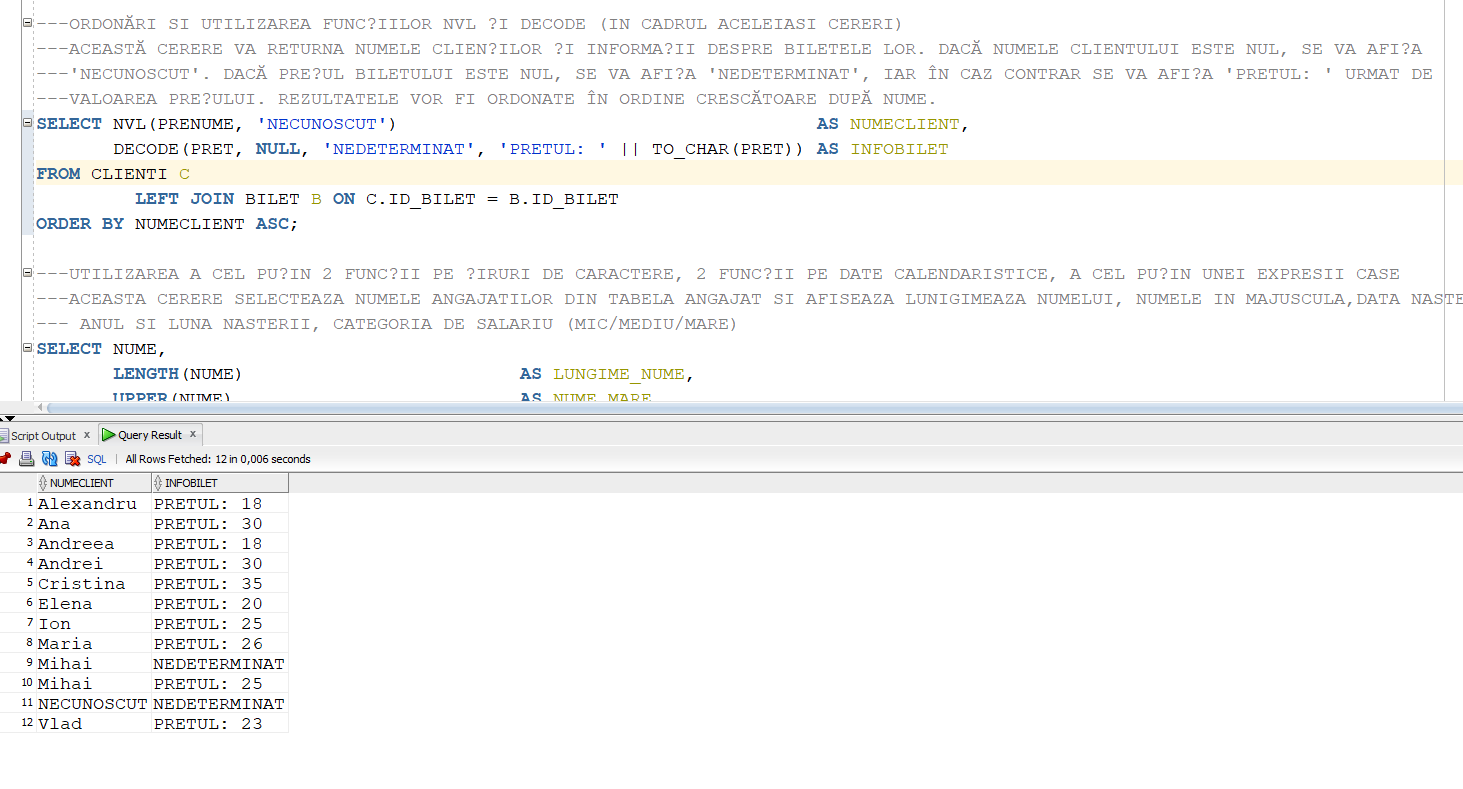
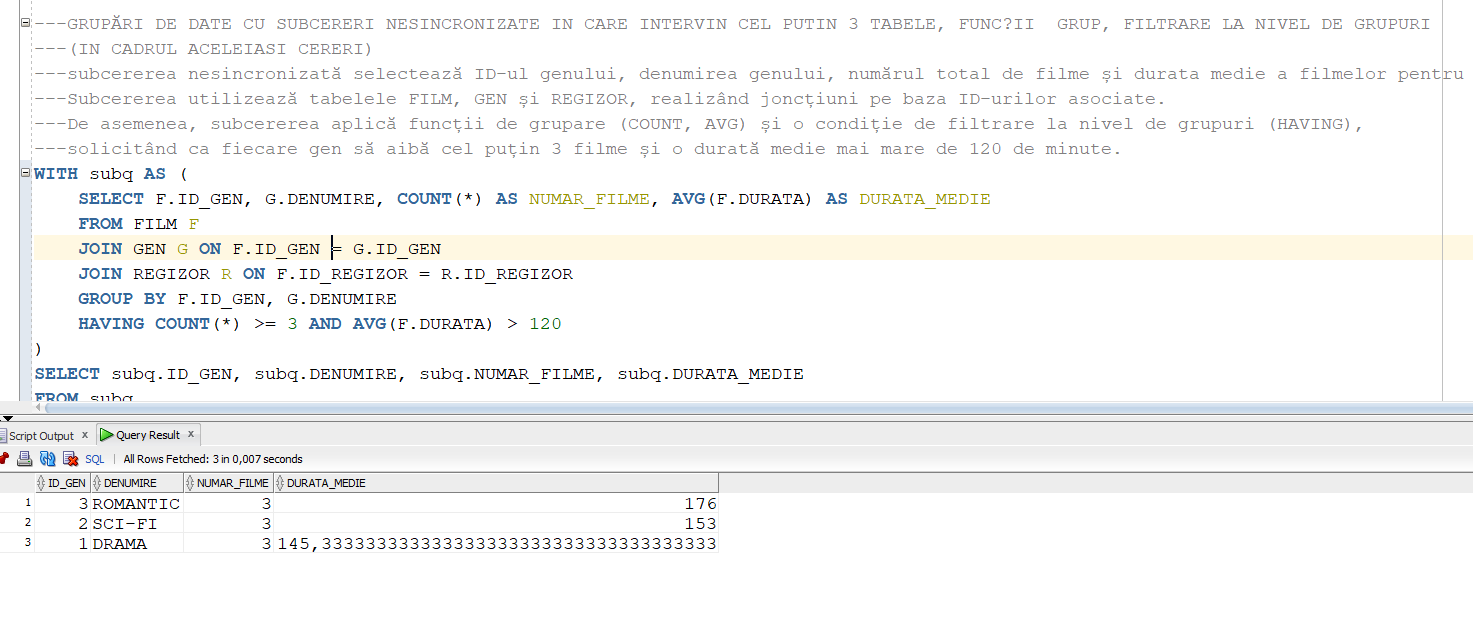
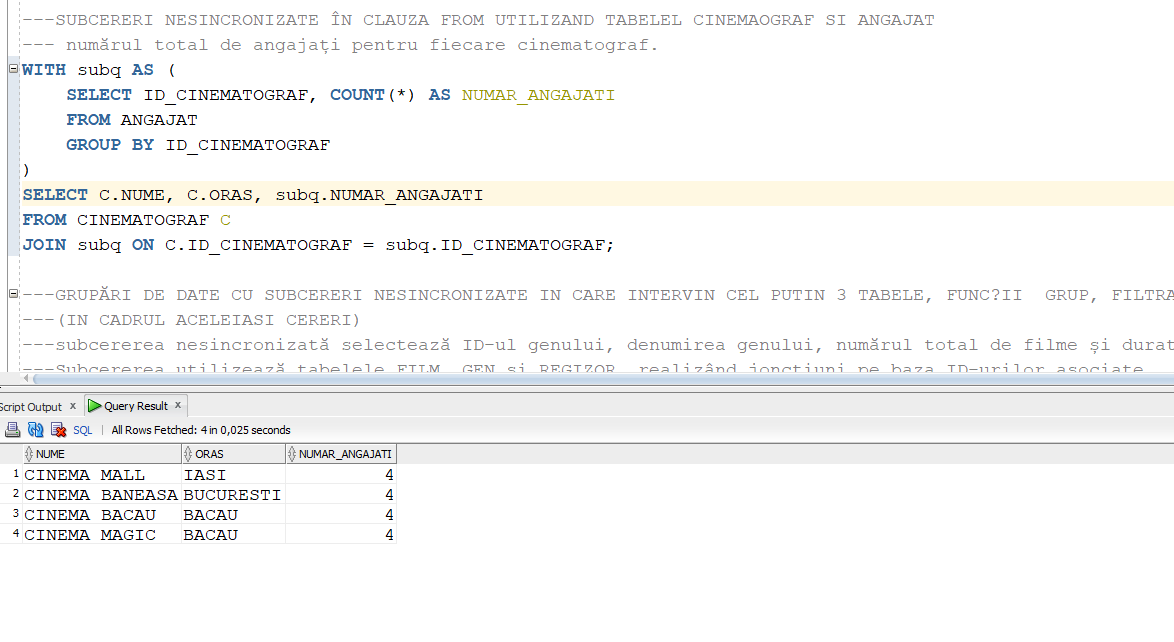
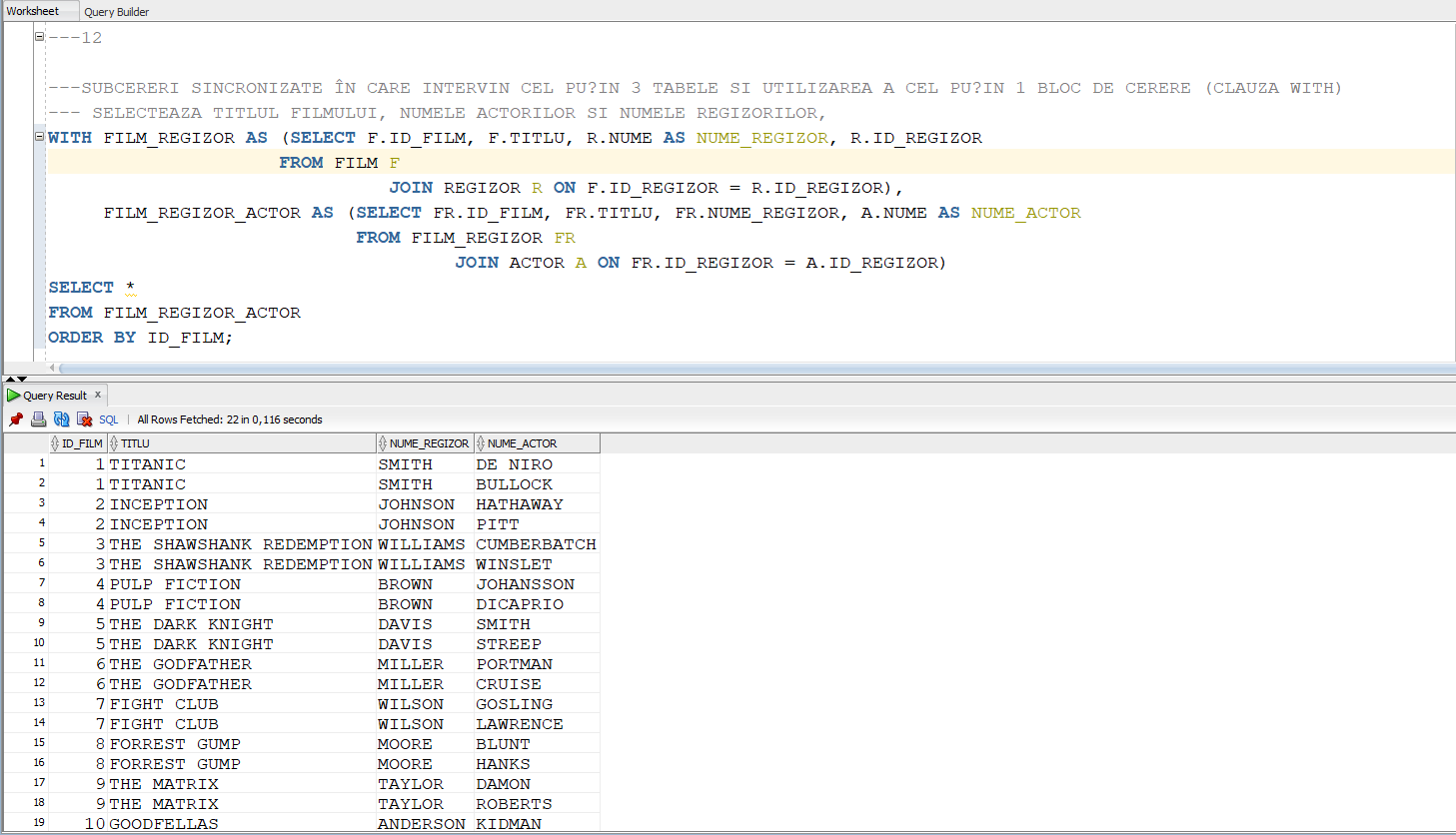
iNSERT INTO CLIENTI (ID\_CLIENT, Nume, Prenume, Data\_Nasterii, Email, Telefon, id\_bilet VALUES (12, 'Tudose', 'Mihai', TO\_DATE('1997-03-25','YYYY-MM-DD'), 'tudose.mihai@gmail.com', '1294582322',11);



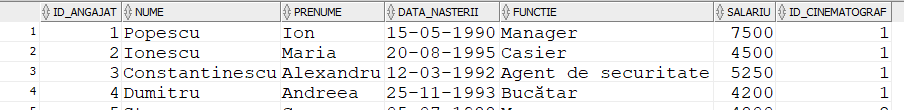
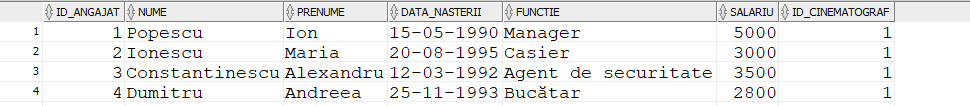
# 12. 5 cereri SQL complexe

Cele 5 cereri vor utiliza in ansamblul lor:

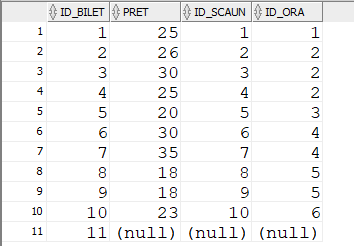
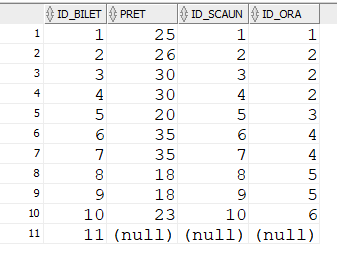
* subcereri sincronizate în care intervin cel puțin 3 tabele
* subcereri nesincronizate în clauza FROM
* grupări de date cu subcereri nesincronizate in care intervin cel putin 3 tabele, funcții grup, filtrare la nivel de grupuri (in cadrul aceleiasi cereri)
* ordonări si utilizarea funcțiilor NVL și DECODE (in cadrul aceleiasi cereri)
* utilizarea a cel puțin 2 funcții pe șiruri de caractere, 2 funcții pe date calendaristice, a cel puțin unei expresii CASE
* utilizarea a cel puțin 1 bloc de cerere (clauza WITH)



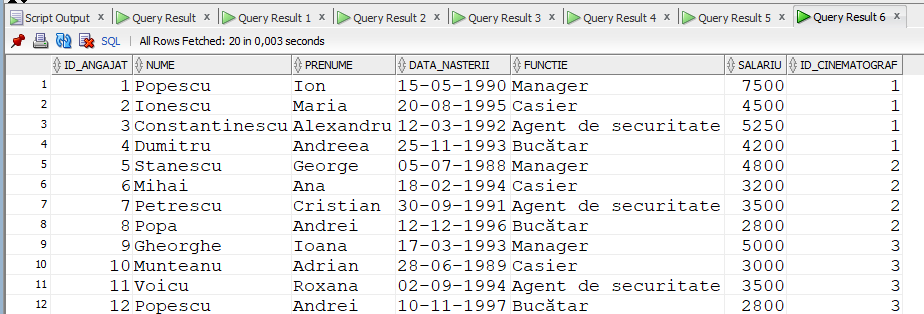
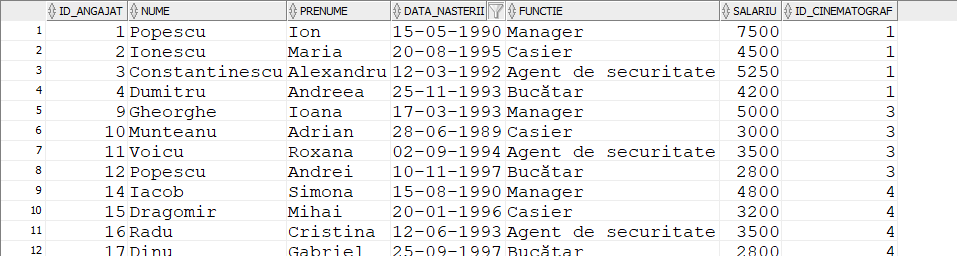
# 13. Implementarea a 3 operații de actualizare și de suprimare a datelor utilizând subcereri

Actualizare salariu pentru angajatii din cinema 1

Reducere pret bilet pt clientii care au ultima litera din prenume I



Stergere angajati din cinema 2



# 14. Crearea unei vizualizări complex

CREATE VIEW VizualizareComplexa AS

SELECT f.ID\_FILM, f.TITLU, r.NUME AS NUME\_REGIZOR, a.NUME AS NUME\_ACTOR

FROM FILM f

JOIN REGIZOR r ON f.ID\_REGIZOR = r.ID\_REGIZOR

JOIN ACTOR a ON r.id\_regizor = a.id\_regizor;

--- Dați un exemplu de operație LMD permisă

SELECT \*

FROM Vizualizare

WHERE NUME\_REGIZOR = 'SMITH';

--- operație LMD nepermisă

UPDATE VizualizareComplexa

SET TITLU = 'New Title'

WHERE ID\_FILM = 1;

Operația UPDATE nu este permisă asupra unei vizualizări complexe deoarece o vizualizare complexă poate include coloane calculate sau combinate din mai multe tabele de bază. Modificarea unei astfel de vedere ar putea duce la o actualizare incorectă sau ambiguă a datelor în tabelele de bază.

# 15. Formulați în limbaj natural și implementați în SQL: o cerere ce utilizează operația outer-join pe minimum 4 tabele, o cerere ce utilizează operația division și o cerere care implementează analiza top-n..

Limbaj natural: Selectați titlul filmului, numele actorului, numele regizorului și denumirea genului pentru toate înregistrările din tabela FILM. Inclusiv, dacă există, afișați numele regizorului și actorului, chiar dacă nu există o potrivire în tabela REGIZOR și respectiv tabela ACTOR. Asigurați-vă că afișați și denumirea genului, indiferent dacă există o potrivire în tabela GEN.

SELECT F.TITLU, A.NUME AS NUME\_ACTOR, R.NUME AS NUME\_REGIZOR, G.DENUMIRE

FROM FILM F

LEFT JOIN REGIZOR R ON F.ID\_REGIZOR = R.ID\_REGIZOR

LEFT JOIN ACTOR A ON R.ID\_REGIZOR = A.ID\_REGIZOR

FULL OUTER JOIN GEN G ON F.ID\_GEN = G.ID\_GEN;

Limbaj Natural: Selectați numele cinematografelor în care toți angajații sunt prezenți. Adică, selectați numele cinematografelor pentru care nu există niciun angajat care să lipsească."

SELECT C.NUME

FROM CINEMATOGRAF C

WHERE NOT EXISTS (

SELECT A.ID\_ANGAJAT

FROM ANGAJAT A

WHERE A.ID\_CINEMATOGRAF = C.ID\_CINEMATOGRAF

AND A.ID\_ANGAJAT NOT IN (

SELECT A2.ID\_ANGAJAT

FROM ANGAJAT A2

WHERE A2.ID\_CINEMATOGRAF = C.ID\_CINEMATOGRAF

)

);

Limbaj natural: Doresc sa obtine primele 2 filme cu rating cel mai mare

SELECT F.ID\_Film, F.Titlu, R.VALOARE

FROM Film F

JOIN RATING R ON F.ID\_RATING=R.ID\_RATING

ORDER BY r.valoare DESC

FETCH FIRST 2 ROWS ONLY;

# 16. Optimizarea unei cereri

Cerere inițială:

SELECT C.NUME, COUNT(A.ID\_ANGAJAT) AS NUMAR\_ANGAJATI

FROM CINEMATOGRAF C

LEFT JOIN ANGAJAT A ON C.ID\_CINEMATOGRAF = A.ID\_CINEMATOGRAF

GROUP BY C.NUME;

Expresia algebrică corespunzătoare acestei cereri este:

γ C.NUME, COUNT(A.ID\_ANGAJAT) → NUMAR\_ANGAJATI (C ⋈ A)

Arborele algebric corespunzător acestei cereri este:

γ C.NUME, COUNT(A.ID\_ANGAJAT) → NUMAR\_ANGAJATI

|

⋈

/ \

C A

| |

ID\_C ID\_C

NUME ID\_ANGAJAT

|

NUME

În acest exemplu, o optimizare potențială ar fi adăugarea unui index pe coloana ID\_CINEMATOGRAF din ambele tabele CINEMATOGRAF și ANGAJAT, pentru a accelera operațiile de join.

Cerere optimizată:

WITH subq AS (

SELECT ID\_CINEMATOGRAF, COUNT(ID\_ANGAJAT) AS NUMAR\_ANGAJATI

FROM ANGAJAT

GROUP BY ID\_CINEMATOGRAF

)

SELECT C.NUME, COALESCE(subq.NUMAR\_ANGAJATI, 0) AS NUMAR\_ANGAJATI

FROM CINEMATOGRAF C

LEFT JOIN subq ON C.ID\_CINEMATOGRAF = subq.ID\_CINEMATOGRA