

UNIVERSITATEA DIN BUCUREȘTI

FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ

DOMENIUL CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI

## **PROIECT LA BAZE DE DATE**

PROFESOR COORDONATOR:

VASILE SILVIU-LAURENȚIU

STUDENT:

CREȚU LAURENȚIU-VASILE

BUCUREȘTI

2022

UNIVERSITATEA DIN BUCUREȘTI

FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ

DOMENIUL CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI

## **MANAGEMENTUL UNEI BIBLIOTECI**

PROFESOR COORDONATOR:

VASILE SILVIU-LAURENȚIU

STUDENT:

CREȚU LAURENȚIU-VASILE

BUCUREȘTI

2022

## CUPRINS

1. Prezentarea modelului.....	6
2. Regulile modelului.....	6
i) AUTORI – CARTI .....	6
ii) EDITURI – CARTI.....	6
iii) CATEGORII – CARTI .....	6
iv) RECENZORI – CARTI .....	6
v) FURNIZORI – CARTI.....	7
vi) CITITORI – CARTI.....	7
3. Diagrama Entitate-Relație.....	8
3.1. Reprezentarea diagramei .....	8
3.2. Descrierea entităților, atributelor, cheilor, relațiilor și a cardinalităților.....	9
3.2.1. Descrierea entităților, atributelor și a cheilor.....	9
3.2.2. Descrierea relațiilor și a cardinalităților:.....	12
4. Diagrama Conceptuală.....	15
4.1. Reprezentarea diagramei .....	15
4.2. Descrierea constrângerilor de integritate.....	16
4.2.1. Tabelul AUTORI: .....	16
4.2.2. Tabelul EDITURI: .....	16
4.2.3. Tabelul CATEGORII:.....	16
4.2.4. Tabelul FURNIZORI: .....	17
4.2.5. Tabelul RECENZORI:.....	17

4.2.6.	Tabelul CITITORI: .....	18
4.2.7.	Tabelul CARTI: .....	18
4.2.8.	Tabelul SCRIE: .....	19
4.2.9.	Tabelul IMPRUMUTA: .....	19
4.2.10.	Tabelul RECENZIE: .....	20
4.3.	Schemele relaționale .....	21
4.3.1.	Schemele relaționale .....	21
4.3.2.	Descrierea constrângerilor ON DELETE .....	21
5.	Scriptul SQL .....	23
5.1.	Introducere .....	23
5.2.	Etapă de DROP .....	23
5.3.	Crearea tabelelor și constrângerilor.....	24
5.3.1.	Tabelul AUTORI: .....	24
5.3.2.	Tabelul EDITURI .....	24
5.3.3.	Tabelul CATEGORII.....	25
5.3.4.	Tabelul FURNIZORI.....	25
5.3.5.	Tabelul RECENZORI.....	26
5.3.6.	Tabelul CITITORI .....	26
5.3.7.	Tabelul CARTI .....	27
5.3.8.	Tabelul SCRIE .....	27
5.3.9.	Tabelul IMPRUMUTA.....	28
5.3.10.	Tabelul RECENZIE .....	28
5.4.	Introducerea datelor în tabele .....	29
5.4.1.	Tabelul AUTORI .....	29
5.4.2.	Tabelul EDITURI .....	29

5.4.3.	Tabelul CATEGORII.....	30
5.4.4.	Tabelul FURNIZORI.....	30
5.4.5.	Tabelul RECENZORI.....	31
5.4.6.	Tabelul CITITORI .....	31
5.4.7.	Tabelul CARTI .....	32
5.4.8.	Tabelul SCRIE .....	32
5.4.9.	Tabelul IMPRUMUTA.....	33
5.4.10.	Tabelul RECENZIE .....	33

## 1. Prezentarea modelului

Tema aleasă pentru proiectul la cursul ‘Baze de date’ este MANAGEMENTUL UNEI BIBLIOTECI. Pentru o bibliotecă, evidența cărților și a împrumuturilor efectuate este foarte importantă, astfel o bază de date corect structurată contează enorm.

Spre exemplu, erorile dintr-o bază de date pot duce la pierderea evidenței împrumuturilor de cărți, din care poate rezulta nerestituirea acestora de către cititori, pierderea datelor cu caracter personal (adrese de e-mail, numere de telefon etc).

Am realizat o bază de date ce are drept scop gestionarea datelor dintr-o bibliotecă. Aceasta stochează date despre cărți (edituri, autori, categorii), date despre furnizori, recenzorii de cărți, cititori și împrumuturile efectuate de către aceștia.

## 2. Regulile modelului

### i) AUTORI – CARTI

- Un autor a scris mai multe cărți, dar există și autori ce nu au scris cărți
- O carte are cel puțin un autor

### ii) EDITURI – CARTI

- O editură corespunde mai multor cărți sau niciuneia
- O carte este publicată de o singură editură

### iii) CATEGORII – CARTI

- O carte aparține unei singure categorii
- O categorie corespunde mai multor cărți, dar pot exista și categorii fără cărți

### iv) RECENZORI – CARTI

- Un recenzor face recenzie cel puțin unei cărți
- O carte este recenzată de mai mulți recenzori sau de niciunul

#### **v) FURNIZORI – CARTI**

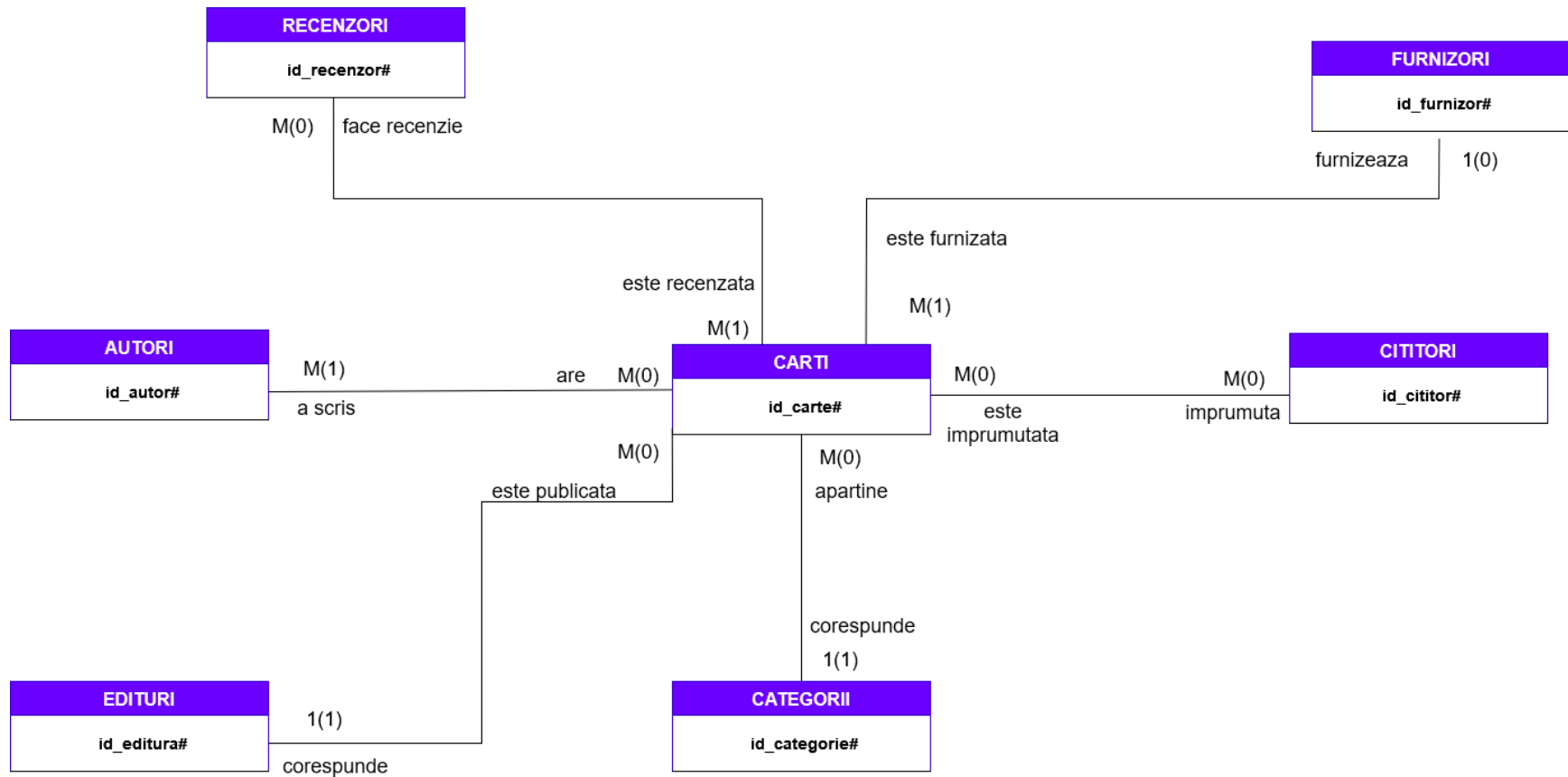
- Un furnizor furnizează cel puțin o carte
- O carte este furnizată de un furnizor sau de niciunul (ex: cărți donate)

#### **vi) CITITORI – CARTI**

- Un cititor poate împrumuta mai multe cărți sau niciuna
- O carte poate fi împrumutată de mai mulți cititori sau de niciunul

### 3. Diagrama Entitate-Relație

#### 3.1.Reprezentarea diagramei





## 3.2. Descrierea entităților, atributelor, cheilor, relațiilor și a cardinalităților

### 3.2.1. Descrierea entităților, atributelor și a cheilor

#### 3.2.1.1. Tabelul *AUTORI*

Tabelul *AUTORI* stochează date importante despre autori în baza de date. Structura tabelului este:

Cheie	Denumire atribut	Tipul de date	Descriere
PK	id_autor	NUMBER(5,0)	Identificatorul unic al unui autor
	nume_autor	VARCHAR2(30 CHAR)	Numele autorului
	prenume_autor	VARCHAR2(30 CHAR)	Prenumele autorului
	data_nasterii	DATE	Data nașterii a unui autor

#### 3.2.1.2. Tabelul *EDITURI*

Tabelul *EDITURI* stochează date despre edituri în baza de date. Structura tabelului este următoarea:

Cheie	Denumire atribut	Tipul de date	Descriere
PK	id_editura	NUMBER(5,0)	Identificatorul unic al unei edituri
	nume_editura	VARCHAR2(30 CHAR)	Numele editurii
	email_editura	VARCHAR2(30 CHAR)	Email-ul de contact al editurii

#### 3.2.1.3. Tabelul *RECENZORI*

Tabelul *RECENZORI* conține informații despre recenzorii de cărți din baza de date. Structura acestui tabel este următoarea:

Cheie	Denumire atribut	Tipul de date	Descriere
PK	id_recenzor	NUMBER(5,0)	Identificatorul unic al unui recenzor
	nume_recenzor	VARCHAR2(30 CHAR)	Numele recenzorului
	prenume_recenzor	VARCHAR2(30 CHAR)	Prenumele recenzorului
	email_recenzor	VARCHAR2(30 CHAR)	Email-ul recenzorului

#### 3.2.1.4. Tabelul CATEGORII

Tabelul CATEGORII stochează date despre categorii în baza de date. Structura tabelului CATEGORII este:

Cheie	Denumire atribut	Tipul de date	Descriere
PK	id_categorie	NUMBER(5,0)	Identificatorul unic al unui categorii
	nume_categorie	VARCHAR2(30 CHAR)	Numele categoriei
	varsta_minima_recomandata	NUMBER(2,0)	Vârsta minimă recomandată pentru un cititor pentru a lectura și împrumuta cărți dintr-o categorie

#### 3.2.1.5. Tabelul FURNIZORI

Tabelul FURNIZORI stochează date despre furnizorii de cărți ai bibliotecii. Structura tabelului este:

Cheie	Denumire atribut	Tipul de date	Descriere
PK	id_furnizor	NUMBER(5,0)	Identificatorul unic al unui furnizor
	nume_furnizor	VARCHAR2(30 CHAR)	Numele furnizorului
	telefon_furnizor	VARCHAR2(10 CHAR)	Numărul de telefon al furnizorului
	email_furnizor	VARCHAR2(30 CHAR)	Email-ul furnizorului

### 3.2.1.6. Tabelul CITITORI

Tabelul CITITORI stochează date despre cititorii ce au abonament la bibliotecă. Structura tabelului este următoarea:

Cheie	Denumire atribut	Tipul de date	Descriere
PK	id_cititor	NUMBER(5,0)	Identificatorul unic al unui cititor
	nume_cititor	VARCHAR2(30 CHAR)	Numele cititorului
	prenume_cititor	VARCHAR2(30 CHAR)	Prenumele cititorului
	telefon_cititor	VARCHAR2(10 CHAR)	Numărul de telefon al cititorului
	email_cititor	VARCHAR2(30 CHAR)	Email-ul cititorului
	data_nasterii	DATE	Data nașterii
	data_inscrierii	DATE	Data la care și-a deschis abonamentul la bibliotecă

### 3.2.1.7. Tabelul CARTI

Tabelul cărți stochează date despre toate cărțile din baza de date. Structura acestuia este următoarea:

Cheie	Denumire atribut	Tipul de date	Descriere
PK	id_carte	NUMBER(7,0)	Identificatorul unic al unei cărți
	titlu	VARCHAR2(50 CHAR)	Titlul cărții
	numar_pagini	NUMBER(4,0)	Numărul de pagini al cărții
	data_publicare	DATE	Data la care cartea a fost publicată
FK	id_editură	NUMBER(5,0)	Id-ul editurii ce a publicat cartea
FK	id_categorie	NUMBER(5,0)	Id-ul categoriei din care face parte cartea
FK	id_furnizor	NUMBER(5,0)	Id-ul furnizorului ce a furnizat cartea

### 3.2.2. Descrierea relațiilor și a cardinalităților:

#### 3.2.2.1. *AUTORI – CARTI*

##### **Relatie:**

- AUTORII au scris CARTI
- CARTILE au AUTORI

##### **Cardinalități:**

- **Cardinalitate maximală**
  - Câți autori are o carte? => MULȚI
  - Câte cărți scrie un autor? => MULTE
- **Cardinalitate minimală**
  - Câți autori trebuie să aibă o carte? => 1
  - Câte cărți trebuie să scrie un autor? => 0

#### 3.2.2.2. *EDITURI – CARTI*

##### **Relatie:**

- EDITURILE corespund CARTILOR
- CARTILE sunt publicate de EDITURI

##### **Cardinalități:**

- **Cardinalitate maximală**
  - O editură câtor cărți corespunde? => MULTORA
  - De câte edituri este publicată o carte? => 1
- **Cardinalitate minimală**
  - O editură câtor cărți trebuie să corespundă? => 0
  - De câte edituri trebuie publicată o carte? => 1

### 3.2.2.3. CATEGORII – CARTI

#### **Relatie:**

- CATEGORIILE corespund CARTILOR
- CARTILE aparțin CATEGORIILOR

#### **Cardinalități:**

- **Cardinalitate maximală**
  - O categorie câtor cărți corespunde? => MULTORA
  - O carte câtor categorii aparține? => 1
- **Cardinalitate minimală**
  - O categorie câtor cărți trebuie să corespundă? => 0
  - O carte câtor categorii trebuie să aparțină? => 1

### 3.2.2.4. RECENZORI – CARTI

#### **Relatie:**

- RECENZORII fac recenzii CARTILOR
- CARTILE sunt recenzate de RECENZORI

#### **Cardinalități:**

- **Cardinalitate maximală**
  - Un recenzor la câte cărți face recenzie? => MULTE
  - De câți recenzori este recenzată o carte? => MULȚI
- **Cardinalitate minimală**
  - Un recenzor la câte cărți trebuie să facă o recenzie? => 1
  - De câți recenzori trebuie să fie recenzată o carte? => 0

### 3.2.2.5. FURNIZORI – CARTI

#### **Relatie:**

- FURNIZORII furnizează CARTI
- CARTILE sunt furnizate de FURNIZORI

#### **Cardinalități:**

- **Cardinalitate maximală**
  - De câți furnizori este furnizată o carte? => 1
  - Câte cărți furnizează un furnizor? => MULTE
- **Cardinalitate minimală**
  - De câți furnizori trebuie să fie furnizată o carte? => 0
  - Câte cărți trebuie să furnizeze un furnizor? => 1

### 3.2.2.6. CITITORI – CARTI

#### **Relatie:**

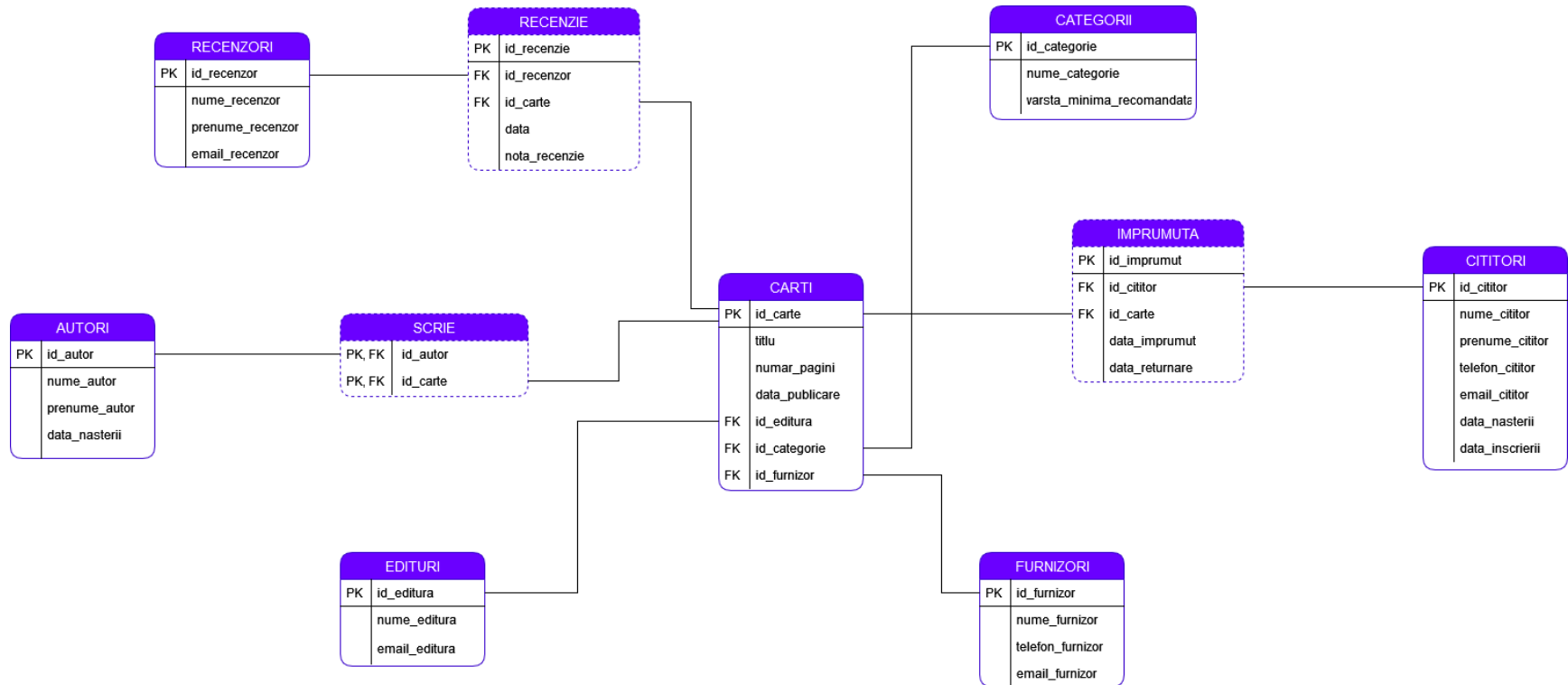
- CITITORII împrumută CARTI
- CARTILE sunt împrumutate de CITITORI

#### **Cardinalități:**

- **Cardinalitate maximală**
  - Câte cărți împrumută un cititor? => MULTE
  - De câți cititori este împrumutată o carte? => MULȚI
- **Cardinalitate minimală**
  - Câte cărți trebuie să împrumute un cititor? => 0
  - De câți cititori trebuie să fie împrumutată o carte? => 0

## 4. Diagrama Conceptuală

### 4.1. Reprezentarea diagramei



## 4.2. Descrierea constrângerilor de integritate

### 4.2.1. Tabelul AUTORI:

- **Constrângeri de tip PRIMARY KEY**
  - AUTORI\_id\_autor\_PK, pentru atributul id\_autor
- **Constrângeri de tip NOT NULL**
  - AUTORI\_ume\_NN, pentru atributul nume\_autor (numele autorului este obligatoriu)
  - AUTORI\_prenume\_NN, pentru atributul prenume\_autor (prenumele autorului este obligatoriu)

### 4.2.2. Tabelul EDITURI:

- **Constrângeri de tip PRIMARY KEY**
  - EDITURI\_id\_editura\_PK, pentru atributul id\_editura
- **Constrângeri de tip NOT NULL**
  - EDITURI\_ume\_NN, pentru atributul nume\_editura (numele editurii este obligatoriu)
  - EDITURI\_email\_NN, pentru atributul email\_editura (email-ul editurii este obligatoriu)
- **Constrângeri de tip UNIQUE**
  - EDITURI\_email\_U, pentru atributul email\_editura (email-ul editurii este unic)

### 4.2.3. Tabelul CATEGORII:

- **Constrângeri de tip PRIMARY KEY**
  - CATEGORII\_id\_categorie\_PK, pentru atributul id\_categorie
- **Constrângeri de tip NOT NULL**
  - CATEGORII\_ume\_NN, pentru atributul nume\_categorie (numele categoriei este obligatoriu)
- **Constrângeri de tip CHECK**
  - CATEGORII\_varsta\_C



- se verifică dacă `varsta_minima_recomandata >= 1`

#### 4.2.4. Tabelul FURNIZORI:

- **Constrângeri de tip PRIMARY KEY**
  - FURNIZORI\_id\_furnizor\_PK, pentru atributul id\_furnizor
- **Constrângeri de tip NOT NULL**
  - FURNIZORI\_nume\_NN, pentru atributul nume\_furnizor (numele furnizorului este obligatoriu)
  - FURNIZORI\_telefon\_NN, pentru atributul telefon\_furnizor (numărul de telefon al unui furnizor este obligatoriu)
  - FURNIZORI\_email\_NN, pentru atributul email\_furnizor (email-ul unui furnizor este obligatoriu)
- **Constrângeri de tip UNIQUE**
  - FURNIZORI\_telefon\_U, pentru atributul telefon\_furnizor (numărul de telefon al unui furnizor este unic)
  - FURNIZORI\_email\_U, pentru atributul email\_furnizor (email-ul unui furnizor este unic)

#### 4.2.5. Tabelul RECENZORI:

- **Constrângeri de tip PRIMARY KEY**
  - RECENZORI\_id\_recenzor\_PK, pentru atributul id\_recenzor
- **Constrângeri de tip NOT NULL**
  - RECENZORI\_nume\_NN, pentru atributul nume\_recenzor (numele unui recenzor este obligatoriu)
  - RECENZORI\_prenume\_NN, pentru atributul prenume\_recenzor (prenumele unui recenzor este obligatoriu)
- **Constrângeri de tip UNIQUE**
  - RECENZORI\_email\_U, pentru atributul email\_recenzor (email-ul unui recenzor este unic)

#### 4.2.6. Tabelul CITITORI:

- **Constrângeri de tip PRIMARY KEY**
  - CITITORI\_id\_cititor\_PK, pentru atributul id\_cititor
- **Constrângeri de tip NOT NULL**
  - CITITORI\_numa\_NN, pentru atributul nume\_cititor (numele unui cititor este obligatoriu)
  - CITITORI\_prenume\_NN, pentru atributul prenume\_cititor (prenumele unui cititor este obligatoriu)
  - CITITORI\_telefon\_NN, pentru atributul telefon\_cititor (numărul de telefon al unui cititor este obligatoriu)
  - CITITORI\_data\_nasterii\_NN, pentru atributul data\_nasterii (data naşterii este obligatorie)
  - CITITORI\_data\_inscrierii\_NN, pentru atributul data\_inscrierii (data înscrierii este obligatorie)
- **Constrângeri de tip UNIQUE**
  - CITITORI\_telefon\_U, pentru atributul telefon\_cititor (numărul de telefon al unui cititor este unic)
  - CITITORI\_email\_U, pentru atributul email\_cititor (email-ul unui cititor este unic)

#### 4.2.7. Tabelul CARTI:

- **Constrângeri de tip PRIMARY KEY**
  - CARTI\_id\_carte\_PK, pentru atributul id\_carte
- **Constrângeri de tip FOREIGN KEY**
  - CARTI\_id\_editura\_FK, pentru atributul id\_editura
    - face legătura cu tabelul EDITURI (id\_editura)
    - determină editura cărţii
  - CARTI\_id\_categorie\_FK, pentru atributul id\_categorie
    - face legătura cu tabelul CATEGORII (id\_categorie)
    - determină categoria din care face parte o carte
  - CARTI\_id\_furnizor\_FK, pentru atributul id\_furnizor

- face legătura cu tabelul FURNIZORI (id\_furnizor)
  - determină furnizorul cărții
- **Constrângeri de tip NOT NULL**
  - CARTI\_titlu\_NN, pentru atributul titlu (titlul unei cărți este obligatoriu)
  - CARTI\_id\_editura\_NN, pentru atributul id\_editura (editura unei cărți este obligatorie)
  - CARTI\_id\_categorie\_NN, pentru atributul id\_categorie (categoria din care face parte o carte este obligatorie)

#### 4.2.8. Tabelul SCRIE:

- **Constrângeri de tip PRIMARY KEY**
  - SCRIE\_PK, pentru attributele (id\_autor, id\_carte)
- **Constrângeri de tip FOREIGN KEY**
  - SCRIE\_id\_autor\_FK, pentru atributul id\_autor
    - face legătura cu tabelul AUTORI (id\_autor)
    - determină autorul ce a scris o serie de cărți
  - SCRIE\_id\_carte\_FK, pentru atributul id\_carte
    - face legătura cu tabelul CARTI (id\_carte)
    - determină cartea căreia îi corespund autori

#### 4.2.9. Tabelul IMPRUMUTA:

- **Constrângeri de tip PRIMARY KEY**
  - IMPRUMUTA\_id\_imprumut\_PK, pentru atributul id\_imprumut
- **Constrângeri de tip FOREIGN KEY**
  - IMPRUMUTA\_id\_cititor\_FK, pentru atributul id\_cititor
    - face legătura cu tabelul CITITORI (id\_cititor)
    - determină cititorul ce a împrumutat cartea
  - IMPRUMUTA\_id\_carte\_FK, pentru atributul id\_carte
    - face legătura cu tabelul CARTI (id\_carte)

- determină cartea ce a fost împrumutată
- **Constrângeri de tip NOT NULL**
  - IMPRUMUTA\_id\_cititor\_NN, pentru atributul id\_cititor (pentru un împrumut cititorul este obligatoriu)
  - IMPRUMUTA\_id\_carte\_NN, pentru atributul id\_carte (cartea este obligatorie)
  - IMPRUMUTA\_data\_imprumut\_NN, pentru atributul data\_imprumut (data la care a fost efectuat împrumutul este obligatorie)

#### 4.2.10. Tabelul RECENZIE:

- **Constrângeri de tip PRIMARY KEY**
  - RECENZIE\_id\_recenzie\_PK, pentru atributul id\_recenzie
- **Constrângeri de tip FOREIGN KEY**
  - RECENZIE\_id\_recenzor\_FK, pentru atributul id\_recenzor
    - face legătura cu tabelul RECENZORI (id\_recenzor)
    - determină recenzorul ce a făcut recenzia
  - RECENZIE\_id\_carte\_FK, pentru atributul id\_carte
    - face legătura cu tabelul CARTI (id\_carte)
    - determină cartea la care face referire recenzia
- **Constrângeri de tip NOT NULL**
  - RECENZIE\_id\_recenzor\_NN, pentru atributul id\_recenzor (o recenzie trebuie să fie scrisă de un recenzor)
  - RECENZIE\_id\_carte\_NN, pentru atributul id\_carte (cartea recenzată este obligatorie)
  - RECENZIE\_data\_NN, pentru atributul data (data la care a fost făcută recenzia este obligatorie)
  - RECENZIE\_nota\_recenzie\_NN, pentru atributul nota\_recenzie (nota cărții în recenzia respectivă este obligatorie)
- **Constrângeri de tip CHECK**
  - RECENZIE\_nota\_recenzie\_C

- verifică dacă nota\_recenzie este mai mare sau egală decât 1 și mai mică sau egală decât 10

### 4.3. Schemele relaționale

#### 4.3.1. Schemele relaționale

Schemele relaționale atașate diagramei conceptuale sunt:

- AUTORI (id\_autor#, nume\_autor, prenume\_autor, data\_nasterii);
- EDITURI (id\_editura#, nume\_editura, email\_editura);
- CATEGORII (id\_categorie#, nume\_categorie, varsta\_minima\_recomandata);
- FURNIZORI (id\_furnizor#, nume\_furnizor, telefon\_furnizor, email\_furnizor);
- RECENZORI (id\_recenzor#, nume\_recenzor, prenume\_recenzor, email\_recenzor);
- CITITORI (id\_cititor#, nume\_cititor, prenume\_cititor, telefon\_cititor, email\_cititor, data\_nasterii, data\_inscrierii);
- CARTI (id\_carte#, titlu, numar\_pagini, data\_publicare, id\_editura (FK), id\_categorie (FK), id\_furnizor (FK));
- SCRIE (id\_autor# (FK), id\_carte# (FK));
- IMPRUMUTA (id\_imprumut#, id\_cititor (FK), id\_carte (FK), data\_imprumut, data\_returnare);
- RECENZIE (id\_recenzie#, id\_recenzor (FK), id\_carte (FK), data, nota\_recenzie);

#### 4.3.2. Descrierea constrângerilor ON DELETE

Există 9 constrângeri ON DELETE, câte una pentru fiecare constrângere de tip FOREIGN KEY. Acestea sunt:

- CARTI\_id\_editura\_FK, **ON DELETE CASCADE**
  - când o editură este ștearsă din baza de date, toate cărțile ce sunt publicate de acea editură vor fi șterse deoarece fiecare carte trebuie să fie publicată de o editură
- CARTI\_id\_categorie\_FK, **ON DELETE CASCADE**
  - când o categorie este ștearsă din baza de date, toate cărțile ce aparțineau acelei categorii vor fi șterse deoarece toate cărțile trebuie să aparțină unei categorii

- CARTI\_id\_furnizor\_FK, **ON DELETE SET NULL**
  - dacă un furnizor este șters din baza de date, cărțile ce au fost furnizate de furnizorul respective vor fi păstrate în baza de date
- SCRIE\_id\_autor\_FK, **ON DELETE CASCADE**
  - în cazul în care un autor este șters din baza de date, cărțile scrise de acel autor vor fi șterse
- SCRIE\_id\_carte\_FK, **ON DELETE CASCADE**
  - în cazul în care o carte este ștearsă din baza de date, atunci toate intrările în care acea carte apare vor fi șterse
- IMPRUMUTA\_id\_cititor\_FK, **ON DELETE CASCADE**
  - în cazul în care un cititor este șters din baza de date, toate împrumuturile asociate acelui cititor vor fi șterse
- IMPRUMUTA\_id\_carte\_FK, **ON DELETE CASCADE**
  - dacă o carte este ștearsă din baza de date, atunci toate împrumuturile acelei cărți vor fi șterse
- RECENZIE\_id\_recenzor\_FK, **ON DELETE CASCADE**
  - dacă un recenzor este șters, atunci toate recenziile scrise de acesta vor fi șterse din baza de date
- RECENZIE\_id\_carte\_FK, **ON DELETE CASCADE**
  - dacă o carte este ștearsă, atunci toate recenziile despre aceasta vor fi șterse

## 5. Scriptul SQL

### 5.1. Introducere

Script-ul SQL, salvat cu numele `SCRIPT_CRETU_LAURENTIU_VASILE_GR261.sql`, a fost scris în Oracle SQL Developer v20, iar pentru baza de date locală am folosit Oracle Database 19c. La prima rulare a script-ului SQL se vor afișa erori datorate operațiilor DROP de la începutul acestuia, dar crearea tabelor și popularea acestora se vor face fără probleme.

### 5.2. Etapa de DROP

Script-ul SQL debutează cu operația de DROP a tabelor și constrângerilor FOREIGN KEY (în tabelele unde există).

```
ALTER TABLE RECENZIE
    DROP CONSTRAINT RECENZIE_id_carte_FK
    DROP CONSTRAINT RECENZIE_id_recenzor_FK;
DROP TABLE RECENZIE;

ALTER TABLE IMPRUMUTA
    DROP CONSTRAINT IMPRUMUTA_id_carte_FK
    DROP CONSTRAINT IMPRUMUTA_id_cititor_FK;
DROP TABLE IMPRUMUTA;

ALTER TABLE SCRIE
    DROP CONSTRAINT SCRIE_id_autor_FK
    DROP CONSTRAINT SCRIE_id_carte_FK;
DROP TABLE SCRIE;

ALTER TABLE CARTI
    DROP CONSTRAINT CARTI_id_categorie_FK
    DROP CONSTRAINT CARTI_id_editura_FK
    DROP CONSTRAINT CARTI_id_furnizor_FK;
DROP TABLE CARTI;
```

```
DROP TABLE CITITORI;  
DROP TABLE RECENZORI;  
DROP TABLE FURNIZORI;  
DROP TABLE CATEGORII;  
DROP TABLE EDITURI;  
DROP TABLE AUTORI;
```

## 5.3. Crearea tabelelor și constrângerilor

### 5.3.1. Tabelul AUTORI:

```
CREATE TABLE AUTORI (  
    id_autor NUMBER(5,0)  
        CONSTRAINT AUTORI_id_autor_PK PRIMARY KEY,  
    nume_autor VARCHAR2(30 CHAR)  
        CONSTRAINT AUTORI_nume_NN NOT NULL,  
    prenume_autor VARCHAR2(30 CHAR)  
        CONSTRAINT AUTORI_prenume_NN NOT NULL,  
    data_nasterii DATE  
);
```

### 5.3.2. Tabelul EDITURI

```
CREATE TABLE EDITURI (  
    id_editura NUMBER(5,0)  
        CONSTRAINT EDITURI_id_editura_PK PRIMARY KEY,  
    nume_editura VARCHAR2(30 CHAR)  
        CONSTRAINT EDITURI_nume_NN NOT NULL,  
    email_editura VARCHAR2(30 CHAR)  
        CONSTRAINT EDITURI_email_NN NOT NULL
```



```
        CONSTRAINT EDITURI_email_U UNIQUE
    );
```

### 5.3.3. Tabelul CATEGORII

```
CREATE TABLE CATEGORII (
    id_categorie NUMBER(5,0)
        CONSTRAINT CATEGORII_id_categorie_PK PRIMARY KEY,
    nume_categorie VARCHAR2(30 CHAR)
        CONSTRAINT CATEGORII_nume_NN NOT NULL,
    varsta_minima_recomandata NUMBER(2,0)
        CONSTRAINT CATEGORII_varsta_C CHECK
(varsta_minima_recomandata >= 1)
);
```

### 5.3.4. Tabelul FURNIZORI

```
CREATE TABLE FURNIZORI (
    id_furnizor NUMBER(5,0)
        CONSTRAINT FURNIZORI_id_furnizor_PK PRIMARY KEY,
    nume_furnizor VARCHAR2(30 CHAR)
        CONSTRAINT FURNIZORI_nume_NN NOT NULL,
    telefon_furnizor VARCHAR2(10 CHAR)
        CONSTRAINT FURNIZORI_telefon_NN NOT NULL
        CONSTRAINT FURNIZORI_telefon_U UNIQUE,
    email_furnizor VARCHAR2(30 CHAR)
        CONSTRAINT FURNIZORI_email_NN NOT NULL
        CONSTRAINT FURNIZORI_email_U UNIQUE
);
```

### 5.3.5. Tabelul RECENZORI

```
CREATE TABLE RECENZORI (  
    id_recenzor NUMBER(5,0)  
        CONSTRAINT RECENZORI_id_recenzor_PK PRIMARY KEY,  
    nume_recenzor VARCHAR2(30 CHAR)  
        CONSTRAINT RECENZORI_nume_NN NOT NULL,  
    prenume_recenzor VARCHAR2(30 CHAR)  
        CONSTRAINT RECENZORI_prenume_NN NOT NULL,  
    email_recenzor VARCHAR2(30 CHAR)  
        CONSTRAINT RECENZORI_email_U UNIQUE  
);
```

### 5.3.6. Tabelul CITITORI

```
CREATE TABLE CITITORI (  
    id_cititor NUMBER(5,0)  
        CONSTRAINT CITITORI_id_cititor_PK PRIMARY KEY,  
    nume_cititor VARCHAR2(30 CHAR)  
        CONSTRAINT CITITORI_nume_NN NOT NULL,  
    prenume_cititor VARCHAR2(30 CHAR)  
        CONSTRAINT CITITORI_prenume_NN NOT NULL,  
    telefon_cititor VARCHAR2(10 CHAR)  
        CONSTRAINT CITITORI_telefon_NN NOT NULL  
        CONSTRAINT CITITORI_telefon_U UNIQUE,  
    email_cititor VARCHAR2(30 CHAR)  
        CONSTRAINT CITITORI_email_U UNIQUE,  
    data_nasterii DATE  
        CONSTRAINT CITITORI_data_nasterii_NN NOT NULL,  
    data_inscrierii DATE  
        CONSTRAINT CITITORI_data_inscrierii_NN NOT NULL  
);
```

### 5.3.7. Tabelul CARTI

```
CREATE TABLE CARTI(  
    id_carte NUMBER(7,0)  
        CONSTRAINT CARTI_id_carte_PK PRIMARY KEY,  
    titlu VARCHAR2(50 CHAR)  
        CONSTRAINT CARTI_titlu_NN NOT NULL,  
    numar_pagini NUMBER(4,0),  
    data_publicare DATE,  
    id_editura NUMBER(5,0)  
        CONSTRAINT CARTI_id_editura_FK REFERENCES  
EDITURI(id_editura) ON DELETE CASCADE  
        CONSTRAINT CARTI_id_editura_NN NOT NULL,  
    id_categorie NUMBER(5,0)  
        CONSTRAINT CARTI_id_categorie_FK REFERENCES  
CATEGORII(id_categorie) ON DELETE CASCADE  
        CONSTRAINT CARTI_id_categorie_NN NOT NULL,  
    id_furnizor NUMBER(5,0)  
        CONSTRAINT CARTI_id_furnizor_FK REFERENCES  
FURNIZORI(id_furnizor) ON DELETE SET NULL  
);
```

### 5.3.8. Tabelul SCRIE

```
CREATE TABLE SCRIE(  
    id_autor NUMBER(5,0)  
        CONSTRAINT SCRIE_id_autor_FK REFERENCES AUTORI(id_autor)  
        ON DELETE CASCADE,  
    id_carte NUMBER(7,0)  
        CONSTRAINT SCRIE_id_carte_FK REFERENCES CARTI(id_carte)  
        ON DELETE CASCADE,
```

```
CONSTRAINT SCRIE_PK PRIMARY KEY(id_autor, id_carte)
);
```

### 5.3.9. Tabelul IMPRUMUTA

```
CREATE TABLE IMPRUMUTA (
    id_imprumut NUMBER(10,0)
        CONSTRAINT IMPRUMUTA_id_imprumut_PK PRIMARY KEY,
    id_cititor NUMBER(5,0)
        CONSTRAINT IMPRUMUTA_id_cititor_FK REFERENCES
CITITORI(id_cititor) ON DELETE CASCADE
        CONSTRAINT IMPRUMUTA_id_cititor_NN NOT NULL,
    id_carte NUMBER(7,0)
        CONSTRAINT IMPRUMUTA_id_carte_FK REFERENCES
CARTI(id_carte) ON DELETE CASCADE
        CONSTRAINT IMPRUMUTA_id_carte_NN NOT NULL,
    data_imprumut DATE
        CONSTRAINT IMPRUMUTA_data_imprumut_NN NOT NULL,
    data_returnare DATE
);
```

### 5.3.10. Tabelul RECENZIE

```
CREATE TABLE RECENZIE (
    id_recenzie NUMBER(8,0)
        CONSTRAINT RECENZIE_id_recenzie_PK PRIMARY KEY,
    id_recenzor NUMBER(5,0)
        CONSTRAINT RECENZIE_id_recenzor_FK REFERENCES
RECENZORI(id_recenzor) ON DELETE CASCADE
        CONSTRAINT RECENZIE_id_recenzor_NN NOT NULL,
    id_carte NUMBER(7,0)
```

```

        CONSTRAINT RECENZIE_id_carte_FK REFERENCES
CARTI(id_carte) ON DELETE CASCADE
        CONSTRAINT RECENZIE_id_carte_NN NOT NULL,
data DATE
        CONSTRAINT RECENZIE_data_NN NOT NULL,
nota_recenzie NUMBER(4,2)
        CONSTRAINT RECENZIE_nota_recenzie_NN NOT NULL
        CONSTRAINT RECENZIE_nota_recenzie_C CHECK
(nota_recenzie >= 1 AND nota_recenzie <= 10)
);

```

## 5.4. Introducerea datelor în tabele

### 5.4.1. Tabelul AUTORI

```

INSERT INTO AUTORI(id_autor, nume_autor, prenume_autor,
data_nasterii) VALUES(1, 'ONCIU', 'ADRIAN', to_date('17-07-
1968','dd-mm-yyyy'));
INSERT INTO AUTORI(id_autor, nume_autor, prenume_autor,
data_nasterii) VALUES(2, 'CHRISTIE', 'AGATHA', to_date('15-09-
1890','dd-mm-yyyy'));
INSERT INTO AUTORI(id_autor, nume_autor, prenume_autor,
data_nasterii) VALUES(3, 'MARTIN-LUGAND', 'AGNES', to_date('22-
07-1979','dd-mm-yyyy'));

```

### 5.4.2. Tabelul EDITURI

```

INSERT INTO EDITURI(id_editura, nume_editura, email_editura)
VALUES(1, 'Act si Politon', 'act.politon@gmail.com');
INSERT INTO EDITURI(id_editura, nume_editura, email_editura)
VALUES(2, 'Ad Libri', 'contact@adlibri.com');
INSERT INTO EDITURI(id_editura, nume_editura, email_editura)

```

```
VALUES(3, 'ALL Educational', 'all@educational.ro');  
INSERT INTO EDITURI(id_editura, nume_editura, email_editura)  
VALUES(4, 'Aramis', 'aramis@gmail.com');
```

#### 5.4.3. Tabelul CATEGORII

```
INSERT INTO CATEGORII(id_categorie, nume_categorie,  
varsta_minima_recomandata) VALUES(1, 'LINGVISTICA', 7);  
INSERT INTO CATEGORII(id_categorie, nume_categorie,  
varsta_minima_recomandata) VALUES(2, 'DICTIONARE', 7);  
INSERT INTO CATEGORII(id_categorie, nume_categorie,  
varsta_minima_recomandata) VALUES(3, 'BIOGRAFII, MEMORII,  
JURNALE', 10);  
INSERT INTO CATEGORII(id_categorie, nume_categorie,  
varsta_minima_recomandata) VALUES(4, 'LIMBI STRAINE', 5);
```

#### 5.4.4. Tabelul FURNIZORI

```
INSERT INTO FURNIZORI(id_furnizor, nume_furnizor,  
telefon_furnizor, email_furnizor) VALUES(1, 'Atlas Group',  
'0702008610', 'contact@atlasgroup.ro');  
INSERT INTO FURNIZORI(id_furnizor, nume_furnizor,  
telefon_furnizor, email_furnizor) VALUES(2, 'Astro Impex',  
'0704892400', 'contact@astro.ro');  
INSERT INTO FURNIZORI(id_furnizor, nume_furnizor,  
telefon_furnizor, email_furnizor) VALUES(3, 'Nicol Cart',  
'0702094430', 'nicol_cart@gmail.com');  
INSERT INTO FURNIZORI(id_furnizor, nume_furnizor,  
telefon_furnizor, email_furnizor) VALUES(4, 'Larry Cart',  
'0785027339', 'larry.cart@gmail.com');
```

#### 5.4.5. Tabelul RECENZORI

```
INSERT INTO RECENZORI(id_recenzor, nume_recenzor,
prenume_recenzor, email_recenzor) VALUES(1, 'WALL', 'FELICIA',
'felicia.wall@gmail.com');
INSERT INTO RECENZORI(id_recenzor, nume_recenzor,
prenume_recenzor, email_recenzor) VALUES(2, 'EWING', 'DANTE',
null);
INSERT INTO RECENZORI(id_recenzor, nume_recenzor,
prenume_recenzor, email_recenzor) VALUES(3, 'BAUER', 'SIDRA',
'sidrabauer21@yahoo.com');
INSERT INTO RECENZORI(id_recenzor, nume_recenzor,
prenume_recenzor, email_recenzor) VALUES(4, 'FOSTER', 'NEHA',
'nehafoster@gmail.com');
```

#### 5.4.6. Tabelul CITITORI

```
INSERT INTO CITITORI(id_cititor, nume_cititor, prenume_cititor,
telefon_cititor, email_cititor, data_nasterii, data_inscrierii)
VALUES(1, 'Ionescu', 'Alin', '0701506075',
'alin.ionescu@gmail.com', to_date('15-05-2003','dd-mm-yyyy'),
to_date('17-03-2019','dd-mm-yyyy'));
INSERT INTO CITITORI(id_cititor, nume_cititor, prenume_cititor,
telefon_cititor, email_cititor, data_nasterii, data_inscrierii)
VALUES(2, 'Popescu', 'Lucian', '0712610752',
'popesculucian54@yahoo.com', to_date('01-10-2001','dd-mm-yyyy'),
to_date('19-10-2018','dd-mm-yyyy'));
INSERT INTO CITITORI(id_cititor, nume_cititor, prenume_cititor,
telefon_cititor, email_cititor, data_nasterii, data_inscrierii)
VALUES(3, 'Antonescu', 'Andreea', '0711185737',
'andreea_anto45@gmail.com', to_date('23-06-1993','dd-mm-yyyy'),
to_date('21-05-2008','dd-mm-yyyy'));
```

```

INSERT INTO CITITORI(id_cititor, nume_cititor, prenume_cititor,
telefon_cititor, email_cititor, data_nasterii, data_inscrierii)
VALUES(4, 'Nehita', 'Florin', '0702039611',
'florinnechitta12@yahoo.com', to_date('06-04-1997','dd-mm-
yyyy'), to_date('03-05-2013','dd-mm-yyyy'));

```

#### 5.4.7. Tabelul CARTI

```

INSERT INTO CARTI(id_carte, titlu, numar_pagini, data_publicare,
id_editura, id_categorie, id_furnizor) VALUES(1, 'Lumea se
destrama', 345, to_date('12-05-1998','dd-mm-yyyy'), 2, 5, 4);
INSERT INTO CARTI(id_carte, titlu, numar_pagini, data_publicare,
id_editura, id_categorie, id_furnizor) VALUES(2, 'Povesti si
povestiri', 213, to_date('13-05-2003','dd-mm-yyyy'), 9, 29, 12);
INSERT INTO CARTI(id_carte, titlu, numar_pagini, data_publicare,
id_editura, id_categorie, id_furnizor) VALUES(3, 'Teoria
grafurilor', 543, to_date('10-03-2003','dd-mm-yyyy'), 23, 13,
1);
INSERT INTO CARTI(id_carte, titlu, numar_pagini, data_publicare,
id_editura, id_categorie, id_furnizor) VALUES(4, 'Matematici
speciale', 693, to_date('21-10-1992','dd-mm-yyyy'), 6, 20, 3);

```

#### 5.4.8. Tabelul SCRIE

```

INSERT INTO SCRIE(id_autor, id_carte) VALUES(3,1);
INSERT INTO SCRIE(id_autor, id_carte) VALUES(9,2);
INSERT INTO SCRIE(id_autor, id_carte) VALUES(20,3);
INSERT INTO SCRIE(id_autor, id_carte) VALUES(35,3);

```



#### 5.4.9. Tabelul IMPRUMUTA

```
INSERT INTO IMPRUMUTA(id_imprumut, id_cititor, id_carte,
data_imprumut, data_returnare) VALUES(1, 1, 10, to_date('17-06-
2020','dd-mm-yyyy'), to_date('30-07-2020','dd-mm-yyyy'));
INSERT INTO IMPRUMUTA(id_imprumut, id_cititor, id_carte,
data_imprumut, data_returnare) VALUES(2, 1, 21, to_date('30-03-
2020','dd-mm-yyyy'), to_date('19-04-2020','dd-mm-yyyy'));
INSERT INTO IMPRUMUTA(id_imprumut, id_cititor, id_carte,
data_imprumut, data_returnare) VALUES(3, 2, 1, to_date('18-02-
2019','dd-mm-yyyy'), to_date('30-03-2019','dd-mm-yyyy'));
INSERT INTO IMPRUMUTA(id_imprumut, id_cititor, id_carte,
data_imprumut, data_returnare) VALUES(4, 4, 41, to_date('20-03-
2015','dd-mm-yyyy'), to_date('17-04-2015','dd-mm-yyyy'));
```

#### 5.4.10. Tabelul RECENZIE

```
INSERT INTO RECENZIE(id_recenzie, id_recenzor, id_carte, data,
nota_recenzie) VALUES(1, 1, 9, to_date('14-03-1998','dd-mm-
yyyy'), 10);
INSERT INTO RECENZIE(id_recenzie, id_recenzor, id_carte, data,
nota_recenzie) VALUES(2, 1, 20, to_date('18-09-1987','dd-mm-
yyyy'), 9.8);
INSERT INTO RECENZIE(id_recenzie, id_recenzor, id_carte, data,
nota_recenzie) VALUES(3, 2, 1, to_date('30-01-2000','dd-mm-
yyyy'), 8.4);
INSERT INTO RECENZIE(id_recenzie, id_recenzor, id_carte, data,
nota_recenzie) VALUES(4, 3, 45, to_date('23-06-2015','dd-mm-
yyyy'), 10);
```