

UNIVERSITATEA DIN BUCUREȘTI

FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ

DEPARTAMENTUL CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI

**PROIECT**  
**BAZE DE DATE**

PROFESOR COORDONATOR:

VASILE SILVIU-LAURENȚIU

STUDENT:

FLORIAN LUCA-PAUL

BUCUREȘTI

2022

UNIVERSITATEA DIN BUCUREȘTI

FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ

DEPARTAMENTUL CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI

## **BAZA DE DATE A UNEI BIBLIOTECI**

PROFESOR COORDONATOR:

VASILE SILVIU-LAURENȚIU

STUDENT:

FLORIAN LUCA-PAUL

BUCUREȘTI

2022

## Cuprins

INTRODUCERE.....	4
PREZENTAREA MODELULUI.....	4
REGULILE MODELULUI.....	5
DIAGRAMA ENTITATE-RELAȚIE.....	6
PREZENTAREA DIAGramei.....	6
DESCRIEREA ENTITĂȚILOR.....	6
DESCRIEREA ATRIBUTELOR.....	7
DESCRIEREA CHEILOR.....	8
DESCRIEREA RELAȚIILOR ȘI A CARDINALITĂȚILOR.....	9
DIAGRAMA CONCEPTUALĂ.....	10
PREZENTAREA DIAGramei.....	10
DESCRIEREA CONSTRÂNGERILOR DE INTEGRITATE.....	11
DESCRIEREA CONSTRÂNGERILOR ON DELETE.....	15
SCHEME RELAȚIONALE.....	16
SCRIPT SQL.....	17
SOFTWARE UTILIZAT.....	17
CONSTRUIREA TABELELOR.....	17
INSERAREA DATELOR.....	21

# INTRODUCERE

## PREZENTAREA MODELULUI

Cărțile, obiectele construite pentru a reține informație în scris, reprezintă una dintre cele mai importante invenții ale lumii: ele ne-au ajutat să ne împărtășim între noi experiențele de viață, lucrurile învățate și, în fond, cunoștințele necesare pentru ca specia noastră să poată evolua.

De-a lungul istoriei, cărțile au fost depozitate în locuri speciale denumite biblioteci, încăperi sau clădiri întregi dedicate exclusiv înmagazinării și protejării materialelor scrise. Pe lângă aceste atribute, multe biblioteci prezintă și avantajul de a fi deschise complet publicului, astfel încât orice persoană dornică de cunoaștere să poată citi direct sau împrumuta aproape orice carte disponibilă în cadrul lor, fără niciun cost suplimentar.

În zilele de astăzi, în care tehnologia prinde un avânt sporit pe zi ce trece, chiar și bibliotecile au nevoie de un sistem informatic robust în spate, menit să ușureze munca și viața atât angajaților ei, cât și a cititorilor cărților sale. Proiectul de față are ca scop construirea unui sistem de acest fel sub forma unei baze de date.

Biblioteca luată drept model în cadrul proiectului prezintă, în principal, două entități importante: cartea și cititorul ce o va putea împrumuta. Cărțile pot fi scrise de unul sau mai mulți autori, pot fi scoase de mai multe edituri, aflate la diverse locații în lume și pot fi grupate, în funcție de genul literar în care sunt scrise, în diverse secțiuni ale bibliotecii, fiecare gestionată de maxim un bibliotecar odată.

În cele ce urmează vor fi prezentate atât regulile, cât și pașii necesari transpunerii modelului respectiv din viața reală într-unul potrivit unei baze de date bine puse la punct.

## REGULILE MODELULUI

Modelul din realitate prezentat anterior este organizat după mai multe reguli:

- Un cititor poate să împrumute una sau mai multe cărți, însă nu este obligatoriu să facă acest lucru
- O carte poate fi împrumutată de unul sau mai mulți cititori, însă ea poate să existe și fără să fie împrumutată
- O carte trebuie să fie scoasă de către o editură, însă ea poate fi scoasă și de mai multe edituri
- O editură poate scoate una sau mai multe ediții ale aceleiași cărți, însă aceasta poate să existe și de sine stătătoare
- O editură poate scoate o singură ediție a unei cărți pe an calendaristic
- O editură trebuie să se afle într-o locație și numai una
- La o singură locație se pot afla una sau mai multe edituri, sau nicio editură
- O carte trebuie să fie scrisă de minim un autor, dar poate fi scrisă și de mai mulți autori
- Un autor poate scrie una sau mai multe cărți, dar poate să existe și fără să scrie o carte
- Prin modul în care este scrisă, o carte se află în maxim o secțiune din cadrul bibliotecii, ce conține strict cărți scrise într-un singur gen literar; ea poate, de altfel, să nu aibă niciun gen specific, și să nu se afle în nicio secțiune
- O secțiune poate avea una sau mai multe cărți, sau poate fi complet goală
- Un bibliotecar trebuie să întrețină obligatoriu o secțiune, dar poate întreține și mai multe secțiuni simultan
- O secțiune este întreținută de maxim un bibliotecar, dar ea poate să existe în cadrul bibliotecii și neîntreținută

*Notă:*

1. În cazul cărților împrumutate de mai mulți cititori simultan, se va presupune că biblioteca are un stoc nelimitat pentru fiecare carte în parte.

# DIAGRAMA ENTITATE-RELATIE

## PREZENTAREA DIAGramei

Un prim pas în proiectarea bazei de date este construirea unei diagrame ce conține atât entitățile principale ale modelului, cât și relațiile dintre ele (*Fig. 1*). Această diagramă are denumirea de diagramă entitate-relație.

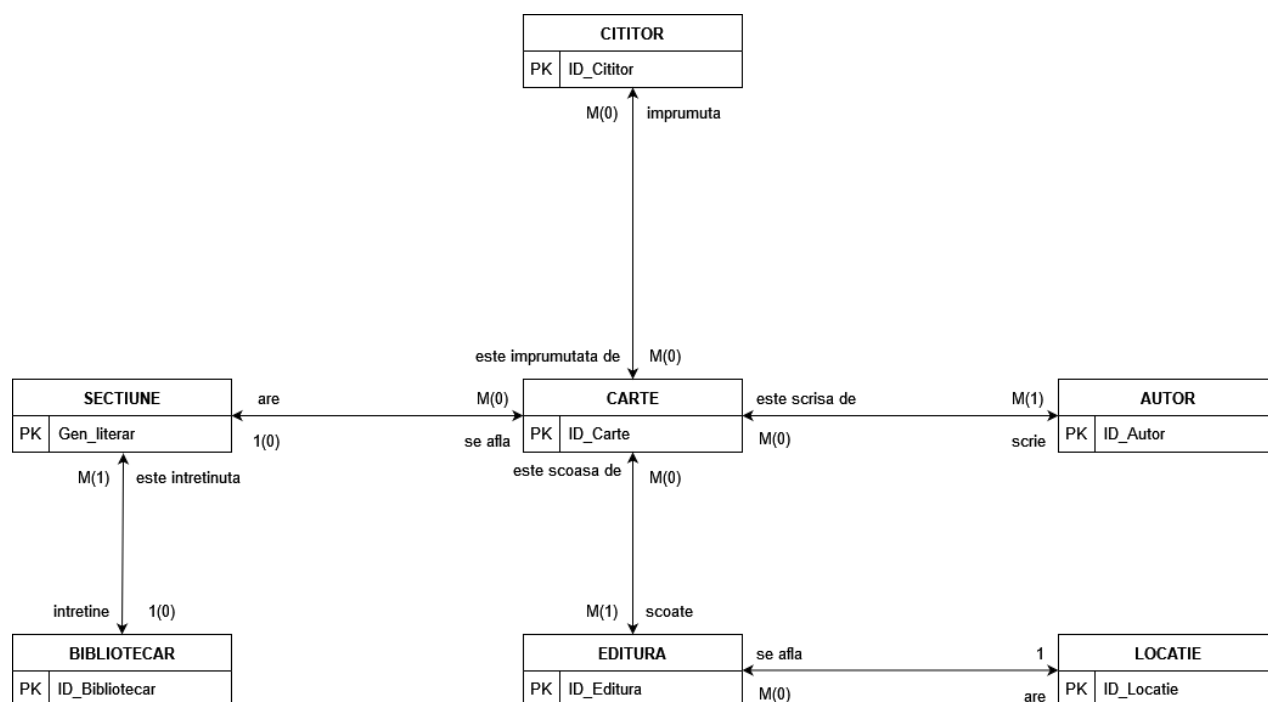


Figura 1 – Diagrama entitate-relație a bazei de date

## DESCRIEREA ENTITĂȚIILOR

Entitățile ce compun modelul sunt:

- **CITITOR** – cititorul care va putea împrumuta cărți în cadrul bibliotecii
- **CARTE** – cartea ce va putea fi împrumutată de către cititori
- **AUTOR** – autorul a una sau mai multe cărți
- **SECTIUNE** – o zonă specifică din interiorul bibliotecii, care conține cărți aparținând unui anumit gen literar
- **BIBLIOTECAR** – angajat al bibliotecii care gestionează cărțile dintr-o anumită secțiune
- **EDITURA** – firmă ce are ca scop publicarea de cărți

- LOCATIE – adresă folosită la localizarea unei edituri

## **DESCRIEREA ATRIBUTELOR**

Atributele entităților prezentate anterior, împreună cu tipul de date prin care vor fi definite în cadrul bazei de date, sunt:

- Pentru entitatea CITITOR:
  - ID\_Cititor – identificator unic al cititorului în cadrul bazei de date – INT
  - Nume – numele cititorului – VARCHAR(20)
  - Prenume – prenumele cititorului – VARCHAR(40)
  - Nr\_telefon – numărul de contact al cititorului – VARCHAR(12)
- Pentru entitatea CARTE:
  - ID\_Carte – identificator unic al cărții în cadrul bazei de date – INT
  - Titlu – titlul cărții – VARCHAR(50)
  - ISBN – Internațional Standard Book Number, identificator unic al cărții în lumea largă – BIGINT
  - Nr\_pagini – numărul de pagini al cărții – INT
  - Gen\_literar – genul literar specific cărții – VARCHAR(20)
- Pentru entitatea AUTOR:
  - ID\_Autor – identificator unic al autorului în cadrul bazei de date – INT
  - Nume – numele autorului – VARCHAR(20)
  - Prenume – prenumele autorului – VARCHAR(40)
  - Data\_nastere – data nașterii autorului – DATE
  - Tara\_origine – numele țării în care s-a născut autorul – VARCHAR(40)
- Pentru entitatea SECTIUNE:
  - Gen\_literar – genul literar al cărților din secțiune, ce identifică unic secțiunea – VARCHAR(20)
  - Nr\_rafturi – numărul de rafturi din secțiune – INT
  - ID\_Bibliotecar – identificatorul unic al bibliotecarului ce întreține secțiunea - INT

- Pentru entitatea BIBLIOTECAR:
  - ID\_Bibliotecar – identificator unic al bibliotecarului în baza de date – INT
  - Nume – numele bibliotecarului – VARCHAR(20)
  - Prenume – prenumele bibliotecarului – VARCHAR(40)
  - Data\_angajare – data la care a fost angajat bibliotecarul – DATE
- Pentru entitatea EDITURA:
  - ID\_Editura – identificator unic al editurii în cadrul bazei de date – INT
  - Nume – numele editurii – VARCHAR(50)
  - ID\_Locatie – identificatorul unic al locației la care se află editura – INT
- Pentru entitatea LOCATIE:
  - ID\_Locatie – identificator unic al unei locații în cadrul bazei de date – INT
  - Tara - țara componentă locației – VARCHAR(40)
  - Oras – orașul component locației – VARCHAR(40)
  - Judet – județul component locației – VARCHAR(30)
  - Strada – strada componentă locației – VARCHAR(50)
  - Numar – numărul locației – VARCHAR(5)

## **DESCRIEREA CHEILOR**

Fiecare entitate are minim o cheie primară ce o identifică unic în cadrul bazei de date:

- Entitatea CITITOR – ID\_Cititor
- Entitatea CARTE – ID\_Carte
- Entitatea AUTOR – ID\_Autor
- Entitatea SECTIUNE – Gen\_literar
- Entitatea BIBLIOTECAR – ID\_Bibliotecar
- Entitatea EDITURA – ID\_Editura
- Entitatea LOCATIE – ID\_Locatie



Pe lângă cheile primare, unele entități prezintă și chei externe, atribute ce participă în constrângeri de integritate cu alte entități (aceste constrângeri sunt detaliate în cadrul descrierii diagramei conceptuale):

- Pentru entitatea CARTE:
  - Gen\_literar – face legătură cu entitatea SECTIUNE
- Pentru entitatea SECTIUNE:
  - ID\_Bibliotecar – face legătură cu entitatea BIBLIOTECAR
- Pentru entitatea EDITURA:
  - ID\_Locatie – face legătură cu entitatea LOCATIE

## **DESCRIEREA RELAȚIILOR ȘI A CARDINALITĂȚILOR**

O relație se poate stabili între oricare două entități ale modelului, fiecare relație având două tipuri de cardinalități: minimală (minimul de instanțe ale unei entități ce pot participa la relație) și maximală (numărul maxim de instanțe ale unei entități care pot participa la relație). Cardinalitățile minimale sunt trecute între paranteze rotunde și sunt adiacente cardinalităților maxime.

În cadrul modelului prezentat, relațiile dintre entități, plus cardinalitățile acestora, sunt:

- Relație M-M (Many-to-many) între CITITOR și CARTE
  - Câți cititori pot împrumuta o carte? – Mulți – M(0)
  - De câți cititori poate fi împrumutată o carte? – Mulți – M(0)
- Relație M-M (Many-to-many) între CARTE și AUTOR
  - Câți autori poate avea o carte? – Mulți, minim unul – M(1)
  - Câte cărți poate scrie un autor? – Multe – M(0)
- Relație M-M (Many-to-many) între CARTE și EDITURA
  - Câte cărți poate scoate o editură? – Multe – M(0)
  - De câte edituri poate fi scoasă o carte? – Multe, minim una – M(1)
- Relație 1-M (One-to-many) între LOCATIE și EDITURA
  - La câte locații trebuie să se afle o editură? – Una – 1

- Câte edituri se pot afla sa o singură locație? – Multe – M(0)
- Relație M-M (Many-to-many) între CARTE și SECTIUNE
  - Câte cărți se pot afla într-o secțiune? – Multe – M(0)
  - În câte secțiuni se poate afla o carte? – Maxim una – 1(0)
- Relație 1-M (One-to-many) între BIBLIOTECAR și SECTIUNE
  - De câți bibliotecari poate fi întreținută o secțiune? – Unul – 1(0)
  - Câte secțiuni poate întreține un bibliotecar? – Multe, minim una – M(1)

*Notă:*

1. Relațiile de tipul Many-to-many vor fi soluționate prin intermediul unor tabele auxiliare în cadrul diagramei conceptuale.
2. S-a omis scrierea cardinalității minimale în cazul în care aceasta este egală cu cea maximală

## **DIAGRAMA CONCEPTUALĂ**

### **PREZENTAREA DIAGramei**

Următorul pas în proiectarea bazei de date este construirea diagramei conceptuale, pe baza căreia se va forma baza de date efectivă (*Fig. 2*). Entitățile devin tabele, attributele devin coloane în cadrul acestor tabele, iar relațiile ce pot induce anomalii logice (cele de tipul Many-to-many) sunt convertite în tabele auxiliare, care sunt legate de cele două tabele originale prin două relații de tipul One-to-many.

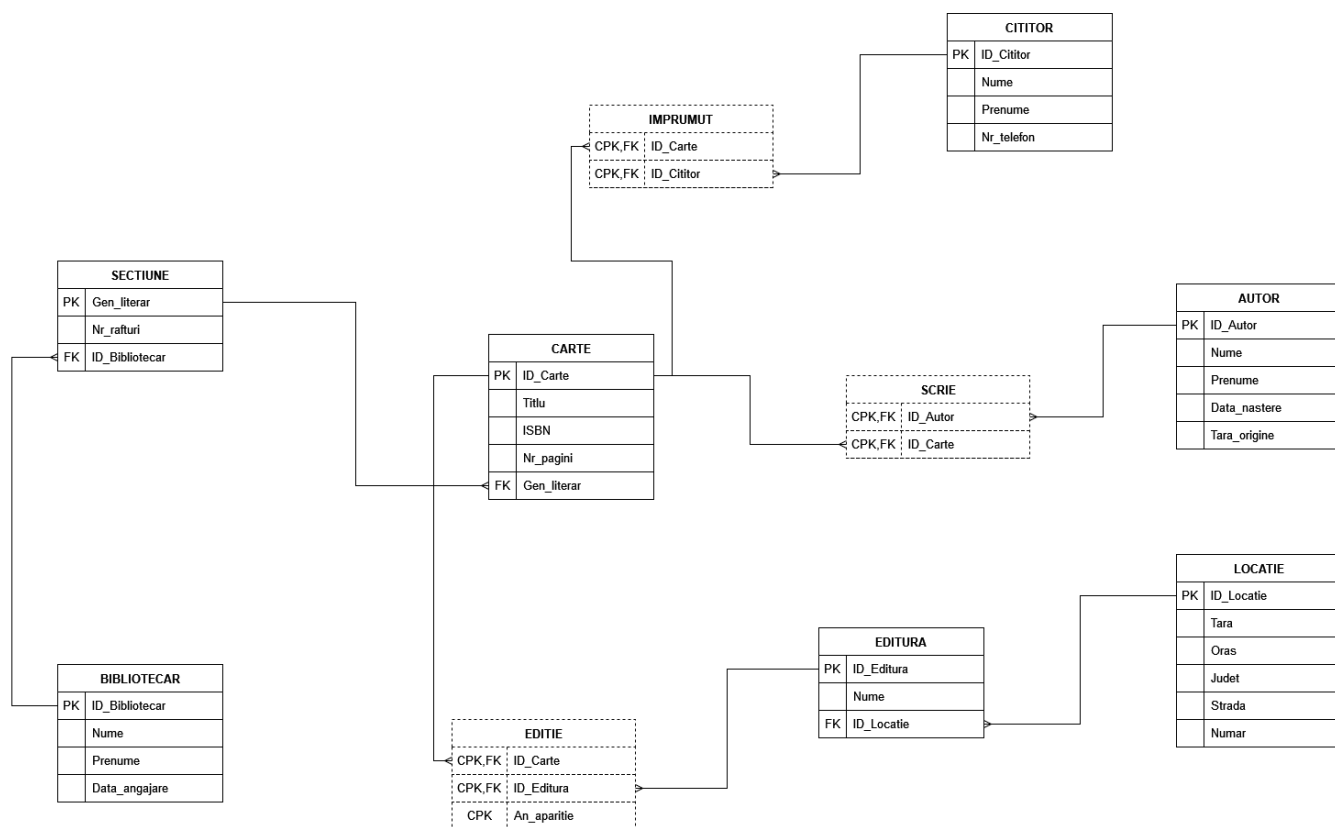


Figura 2 – Diagrama conceptuală a bazei de date

## DESCRIEREA CONSTRÂNGERILOR DE INTEGRITATE

Constrângerile de integritate reprezintă un set de reguli implementate bazei de date ce permit buna funcționare a acesteia și păstrarea calității informației conținută înăuntrul său.

Fiecare tabel are propriile constrângeri de integritate, cele mai importante dintre ele având câte un nume unic în cadrul bazei de date, în scopul de a putea fi gestionate ușor de către administratorul acesteia.

Constrângerile tabelelor principale ale bazei de date a bibliotecii sunt:

- Tabelul CITITOR:
  - Constrângere de tip CHEIE PRIMARĂ:
    - PK\_Cititor - Valorile coloanei *ID\_Cititor*, asociată cheii primare a tabelului, trebuie să fie unice și nenule
  - Constrângeri de tip NOT NULL:

- Valorile coloanei *Nume* trebuie să fie nenule
  - Valorile coloanei *Prenume* trebuie să fie nenule
- Tabelul CARTE:
  - Constrângere de tip CHEIE PRIMARĂ:
    - PK\_Carte – Valorile coloanei *ID\_Carte*, asociată cheii primare a tabelului, trebuie să fie unice și nenule
  - Constrângere de tip CHEIE EXTERNĂ:
    - FK\_Carte\_Sectiune – Valorile coloanei *Gen\_literar*, asociată cheii externe a tabelului, trebuie să fie ori nule, ori să corespundă unor valori ale cheii primare din tabelul SECTIUNE la care face referire cheia externă
  - Constrângere de tip NOT NULL:
    - Valorile coloanei *Titlu* trebuie să fie nenule
  - Constrângere de tip CHECK:
    - CHK\_Carte\_Nr\_pagini\_Poz - Valorile coloanei *Nr\_pagini* trebuie să fie pozitive
- Tabelul AUTOR:
  - Constrângere de tip CHEIE PRIMARĂ:
    - PK\_Autor – Valorile coloanei *ID\_Autor*, asociată cheii primare a tabelului, trebuie să fie unice și nenule
  - Constrângeri de tip NOT NULL:
    - Valorile coloanei *Nume* trebuie să fie nenule
    - Valorile coloanei *Prenume* trebuie să fie nenule
- Tabelul SECTIUNE:
  - Constrângere de tip CHEIE PRIMARĂ:
    - PK\_Sectiune - Valorile coloanei *Gen\_literar*, asociată cheii primare a tabelului, trebuie să fie unice și nenule
  - Constrângere de tip CHEIE EXTERNĂ:

- FK\_Sectiune\_Bibliotecar – Valorile coloanei *ID\_Bibliotecar*, asociată cheii externe a tabelului, trebuie să fie ori nule, ori să corespundă unor valori ale cheii primare din tabelul BIBLIOTECAR la care face referire cheia externă
  - Constrângere de tip CHECK:
    - CHK\_Sectiune\_Nr\_rafturi\_Poz – Valorile coloanei *Nr\_rafturi* trebuie să fie pozitive
- Tabelul BIBLIOTECAR:
  - Constrângere de tip CHEIE PRIMARĂ:
    - PK\_Bibliotecar – Valorile coloanei *ID\_Bibliotecar*, asociată cheii primare a tabelului, trebuie să fie unice și nenule
  - Constrângeri de tip NOT NULL:
    - Valorile coloanei *Nume* trebuie să fie nenule
    - Valorile coloanei *Prenume* trebuie să fie nenule
    - Valorile coloanei *Data\_angajare* trebuie să fie nenule
- Tabelul EDITURA:
  - Constrângere de tip CHEIE PRIMARĂ:
    - PK\_Editura – Valorile coloanei *ID\_Editura*, asociată cheii primare a tabelului, trebuie să fie unice și nenule
  - Constrângere de tip CHEIE EXTERNĂ:
    - FK\_Editura\_Locatie – Valorile coloanei *ID\_Locatie*, asociată cheii externe a tabelului, trebuie să corespundă unor valori ale cheii primare din tabelul LOCATIE la care face referire cheia externă
  - Constrângeri de tip NOT NULL:
    - Valorile coloanei *Nume* trebuie să fie nenule
    - Valorile coloanei *ID\_Locatie* trebuie să fie nenule
- Tabelul LOCATIE:
  - Constrângere de tip CHEIE PRIMARĂ:
    - PK\_Locatie – Valorile coloanei *ID\_Locatie*, asociată cheii primare a tabelului, trebuie să fie unice și nenule

- Constrângeri de tip NOT NULL:
  - Valorile coloanei *Tara* trebuie să fie nenule
  - Valorile coloanei *Oras* trebuie să fie nenule

Constrângerile tabelelor auxiliare ale bazei de date sunt:

- Tabelul IMPRUMUT:
  - Constrângere de tip CHEIE PRIMARĂ:
    - CPK\_Imprumut – Valorile coloanelor *ID\_Carte* și *ID\_Cititor*, asociate cheii primare a tabelului, trebuie să fie unice și nenule
  - Constrângeri de tip CHEIE EXTERNĂ:
    - FK\_Imprumut\_Carte – Valorile coloanei *ID\_Carte*, asociată unei chei externe a tabelului, trebuie să corespundă unor valori ale cheii primare din tabelul CARTE la care face referire cheia externă
    - FK\_Imprumut\_Cititor – Valorile coloanei *ID\_Cititor*, asociată unei chei externe a tabelului, trebuie să corespundă unor valori ale cheii primare din tabelul CITITOR la care face referire cheia externă
- Tabelul SCRIE:
  - Constrângere de tip CHEIE PRIMARĂ:
    - CPK\_Scrie – Valorile coloanelor *ID\_Carte* și *ID\_Autor*, asociate cheii primare a tabelului, trebuie să fie unice și nenule
  - Constrângeri de tip CHEIE EXTERNĂ:
    - FK\_Scrie\_Carte – Valorile coloanei *ID\_Carte*, asociată unei chei externe a tabelului, trebuie să corespundă unor valori ale cheii primare din tabelul CARTE la care face referire cheia externă
    - FK\_Scrie\_Autor – Valorile coloanei *ID\_Autor*, asociată unei chei externe a tabelului, trebuie să corespundă unor valori ale cheii primare din tabelul AUTOR la care face referire cheia externă
- Tabelul EDITIE:
  - Constrângere de tip CHEIE PRIMARĂ:

- CPK\_Editie – Valorile coloanelor *ID\_Carte*, *ID\_Editura* și *An\_aparitie*, asociate cheii primare a tabelului, trebuie să fie unice și nenule
- Constrângeri de tip CHEIE EXTERNĂ:
  - FK\_Editie\_Carte – Valorile coloanei *ID\_Carte*, asociată unei chei externe a tabelului, trebuie să corespundă unor valori ale cheii primare din tabelul CARTE la care face referire cheia externă
  - FK\_Editie\_Editura – Valorile coloanei *ID\_Editura*, asociată unei chei externe a tabelului, trebuie să corespundă unor valori ale cheii primare din tabelul EDITURA la care face referire cheia externe

## **DESCRIEREA CONSTRÂNGERILOR ON DELETE**

Constrângerile de tip ON DELETE descriu comportamentul cheilor externe sau a rândurilor dintr-un tabel, la ștergerea rândurilor din tabelul la care se face referire.

Fiecare dintre cheile externe ale tabelelor bazei de date au câte o constrângere de tip ON DELETE, și anume:

- Tabelul CARTE:
  - FK\_Carte\_Sectiune are constrângerea ON DELETE SET NULL – dacă se desființează o secțiune din cadrul bibliotecii, cărțile aparținând acelui gen literar vor fi considerate drept cărți fără un gen literar specific
- Tabelul SECTIUNE:
  - FK\_Sectiune\_Bibliotecar are constrângerea ON DELETE SET NULL – dacă un bibliotecar nu mai face parte din bibliotecă, secțiunile întreținute de acesta vor deveni neîntreținute
- Tabelul EDITURA:
  - FK\_Editura\_Locatie are constrângerea ON DELETE CASCADE – o editură nu poate exista fără o locație aferentă

Același principiu se aplică și tabelelor auxiliare:

- Tabelul IMPRUMUT:

- FK\_Imprumut\_Carte are constrângerea ON DELETE CASCADE – un împrumut nu poate exista fără o carte
- FK\_Imprumut\_Cititor are constrângerea ON DELETE CASCADE – un împrumut nu poate exista fără un cititor
- Tabelul SCRIE:
  - FK\_Scrie\_Carte are constrângerea ON DELETE CASCADE – dacă este scoasă o carte din bibliotecă, este ștearsă și intrarea asociată ei din tabelul auxiliar
  - FK\_Scrie\_Autor are constrângerea ON DELETE CASCADE – dacă un autor este ștears, sunt șterse din tabelul auxiliar intrările tuturor cărților asociate autorului respectiv
- Tabelul EDITIE:
  - FK\_Editie\_Carte are constrângerea ON DELETE CASCADE – dacă este scoasă o carte din bibliotecă, sunt șterse toate edițiile aferente ei din tabelul auxiliar
  - FK\_Editie\_Editura are constrângerea ON DELETE CASCADE – dacă este ștearsă o editură, sunt șterse din tabelul auxiliar toate intrările asociate editurii respective

## **SCHEME RELAȚIONALE**

Schemele relaționale ale diagramei conceptuale sunt:

- CITITOR (ID\_Cititor (PK), Nume, Prenume, Nr\_telefon)
- CARTE (ID\_Carte (PK), Titlu, ISBN, Nr\_pagini, Gen\_literar (FK))
- AUTOR (ID\_Autor (PK), Nume, Prenume, Data\_nastere, Tara\_origine)
- SECTIUNE (Gen\_literar, Nr\_rafturi, ID\_Bibliotecar (FK))
- BIBLIOTECAR (ID\_Bibliotecar (PK), Nume, Prenume, Data\_angajare)
- EDITURA (ID\_Editura (PK), Nume, ID\_Locatie (FK))
- LOCATIE (ID\_Locatie (PK), Tara, Oras, Judet, Strada, Numar)
- IMPRUMUT (ID\_Carte (CPK, FK), ID\_Cititor (CPK, FK))
- SCRIE (ID\_Carte (CPK, FK), ID\_Autor(CPK, FK))
- EDITIE (ID\_Carte (CPK, FK), ID\_Editura (CPK, FK), An\_aparitiei (CPK))



## **SCRIPT SQL**

### **SOFTWARE UTILIZAT**

Crearea atât a tabelelor, cât și ale coloanelor și ale constrângerilor de integritate ale acestora a fost făcută cu ajutorul SGBD-ului (Sistem de Gestiune a Bazelor de Date) MySQL v8.0.27, folosind instrucțiuni aparținând limbajului standard SQL (Structured Query Language).

### **CONSTRUIREA TABELELOR**

Instrucțiunile utilizate pentru construirea tabelelor detaliate anterior sunt următoarele:

```
CREATE TABLE Bibliotecar(
```

```
    ID_Bibliotecar INT,
```

```
    Nume VARCHAR(20) NOT NULL,
```

```
    Prenume VARCHAR(40) NOT NULL,
```

```
    Data_angajare DATE NOT NULL,
```

```
    CONSTRAINT PK_Bibliotecar PRIMARY KEY(ID_Bibliotecar)
```

```
);
```

```
CREATE TABLE Sectiune(
```

```
    Gen_literar VARCHAR(20),
```

```
    Nr_rafturi INT,
```

```
    ID_Bibliotecar INT,
```

```
    CONSTRAINT PK_Sectiune PRIMARY KEY(Gen_literar),
```

```
    CONSTRAINT FK_Sectiune_Bibliotecar FOREIGN KEY(ID_Bibliotecar)
REFERENCES bibliotecar(ID_Bibliotecar) ON DELETE SET NULL,
```

```
    CONSTRAINT CHK_Sectiune_Nr_rafturi_Poz CHECK (Nr_rafturi>0)
```

```
);
```

```

CREATE TABLE Carte(
    ID_Carte INT,
    Titlu VARCHAR(50) NOT NULL,
    ISBN BIGINT,
    Nr_pagini INT,
    Gen_literar VARCHAR(20),
    CONSTRAINT PK_Carte PRIMARY KEY(ID_Carte),
    CONSTRAINT FK_Carte_Sectiune FOREIGN KEY(Gen_literar) REFERENCES
sectiune(Gen_literar) ON DELETE SET NULL,
    CONSTRAINT CHK_Carte_Nr_pagini_Poz CHECK (Nr_pagini>0)
);

```

```

CREATE TABLE Cititor(
    ID_Cititor INT,
    Nume VARCHAR(20) NOT NULL,
    Prenume VARCHAR(40) NOT NULL,
    Nr_telefon VARCHAR(12),
    CONSTRAINT PK_Cititor PRIMARY KEY(ID_Cititor)
);

```

```

CREATE TABLE Imprumut(
    ID_Carte INT,
    ID_Cititor INT,
    CONSTRAINT CPK_Imprumut PRIMARY KEY(ID_Carte,ID_Cititor),

```

```
        CONSTRAINT FK_Imprumut_Carte FOREIGN KEY(ID_Carte) REFERENCES  
carte(ID_Carte) ON DELETE CASCADE,
```

```
        CONSTRAINT FK_Imprumut_Cititor FOREIGN KEY(ID_Cititor) REFERENCES  
cititor(ID_Cititor) ON DELETE CASCADE
```

```
);
```

```
CREATE TABLE Autor(  
    ID_Autor INT,
```

```
    Nume VARCHAR(20) NOT NULL,
```

```
    Prenume VARCHAR(40) NOT NULL,
```

```
    Data_nastere DATE,
```

```
    Tara_origine VARCHAR(40),
```

```
    CONSTRAINT PK_Autor PRIMARY KEY(ID_Autor)
```

```
);
```

```
);
```

```
CREATE TABLE Srie(  
    ID_Carte INT,
```

```
    ID_Autor INT,
```

```
    CONSTRAINT CPK_Srie PRIMARY KEY(ID_Carte, ID_Autor),
```

```
    CONSTRAINT FK_Srie_Carte FOREIGN KEY(ID_Carte) REFERENCES
```

```
carte(ID_Carte) ON DELETE CASCADE,
```

```
    CONSTRAINT FK_Srie_Autor FOREIGN KEY(ID_Autor) REFERENCES  
autor(ID_Autor) ON DELETE CASCADE
```

```
);
```

```
CREATE TABLE Locatie(  
    ID_Locatie INT,
```

```

        ID_Locatie INT,

        Tara VARCHAR(40) NOT NULL,

        Oras VARCHAR(40) NOT NULL,

        Judet VARCHAR(30),

        Strada VARCHAR(50),

        Numar VARCHAR(5),

        CONSTRAINT PK_Locatie PRIMARY KEY(ID_Locatie)

);

CREATE TABLE Editura(

        ID_Editura INT,

        Nume VARCHAR(50) NOT NULL,

        ID_Locatie INT NOT NULL,

        CONSTRAINT PK_Editura PRIMARY KEY(ID_Editura),

        CONSTRAINT FK_Editura_Locatie FOREIGN KEY(ID_Locatie) REFERENCES
locatie(ID_Locatie) ON DELETE CASCADE

);

CREATE TABLE Editie(

        ID_Carte INT,

        ID_Editura INT,

        An_aparitiei INT,

        CONSTRAINT CPK_Editie PRIMARY KEY(ID_Carte,ID_Editura,An_aparitiei),

```

```
CONSTRAINT FK_Editie_Carte FOREIGN KEY(ID_Carte) REFERENCES  
carte(ID_Carte) ON DELETE CASCADE,
```

```
CONSTRAINT FK_Editie_Editura FOREIGN KEY(ID_Editura) REFERENCES  
editura(ID_Editura) ON DELETE CASCADE
```

```
);
```

Înainte de crearea tabelor, în script-ul SQL au fost introduse o serie de instrucțiuni de tip DROP menite să șteargă toate tabelele, în cazul în care acestea există deja. Scopul acestei metode este acela de a permite utilizatorului să ruleze același script de generare a tabelor de mai multe ori, fără să întâmpine erori:

```
DROP TABLE IF EXISTS imprumut;
```

```
DROP TABLE IF EXISTS scie;
```

```
DROP TABLE IF EXISTS editie;
```

```
DROP TABLE IF EXISTS cititor;
```

```
DROP TABLE IF EXISTS autor;
```

```
DROP TABLE IF EXISTS carte;
```

```
DROP TABLE IF EXISTS sectiune;
```

```
DROP TABLE IF EXISTS bibliotecar;
```

```
DROP TABLE IF EXISTS editura;
```

```
DROP TABLE IF EXISTS locatie;
```

## **INSERAREA DATELOR**

Cu ajutorul unor instrucțiuni de inserție a datelor, tabelele generate anterior au fost populate cu o serie de date alese arbitrar, pentru a se putea verifica dacă baza de date funcționează corespunzător și respectă toate regulile impuse.

Ordinea inserării a fost aleasă în felul următor: prima oară se introduc date în tabelele fără chei externe (i.e. fără legături cu alte tabele), apoi în restul tabelor.

Pentru tabelul BIBLIOTECAR:

```
INSERT INTO Bibliotecar VALUES(1, 'Esteră', 'Constantin', '2014-03-28');
```

```
INSERT INTO Bibliotecar VALUES(2, 'Ilinca', 'Petrescu', '2006-07-04');
```

```
INSERT INTO Bibliotecar VALUES(3, 'Miron', 'Albert', '2020-11-15');
```

```
INSERT INTO Bibliotecar VALUES(4, 'Stelian', 'Iliescu', '2008-05-19');
```

```
INSERT INTO Bibliotecar VALUES(5, 'Damian', 'Negrescu', '1999-06-01');
```

```
INSERT INTO Bibliotecar VALUES(6, 'Theodor', 'Matei', '2018-04-04');
```

```
INSERT INTO Bibliotecar VALUES(7, 'Emanuela', 'Funar', '2004-09-12');
```

```
INSERT INTO Bibliotecar VALUES(8, 'Stelian', 'Iliescu', '2000-05-19');
```

Pentru tabelul SECTIUNE:

```
INSERT INTO Sectiune VALUES('Crimă, Mister', 10, 1);
```

```
INSERT INTO Sectiune VALUES('Thriller, Horror', 15, 2);
```

```
INSERT INTO Sectiune VALUES('Aventura', 30, 3);
```

```
INSERT INTO Sectiune VALUES('Science Fiction', 25, 4);
```

```
INSERT INTO Sectiune VALUES('Fantasy', 20, 5);
```

```
INSERT INTO Sectiune VALUES('Romance', 8, 6);
```

```
INSERT INTO Sectiune VALUES('Istorie', 23, 7);
```

```
INSERT INTO Sectiune VALUES('Biografie', 12, 8);
```

```
INSERT INTO Sectiune VALUES('Jurnalism', 10, 6);
```

```
INSERT INTO Sectiune VALUES('Filosofie', 5, null);
```

Pentru tabelul CARTE:

INSERT INTO Carte VALUES(1, 'Moarte printre nori', 9786063380464, 320, 'Crimă, Mister');

INSERT INTO Carte VALUES(2, 'Crima din Orient Express', 9786063375477, 288, 'Crimă, Mister');

INSERT INTO Carte VALUES(3, 'Memoriile lui Sherlock Holmes', 9786063378836, 400, 'Crimă, Mister');

INSERT INTO Carte VALUES(4, 'O minciună perfectă', 9786063373909, 368, 'Crimă, Mister');

INSERT INTO Carte VALUES(5, 'IT', 9781473666931, 1392, 'Thriller, Horror');

INSERT INTO Carte VALUES(6, 'Jocul', 9786063378935, 432, 'Thriller, Horror');

INSERT INTO Carte VALUES(7, 'Povestea lui Lisey', 9786064311801, 608, 'Thriller, Horror');

INSERT INTO Carte VALUES(8, 'Billy Summers', 9786064312204, 608, 'Thriller, Horror');

INSERT INTO Carte VALUES(9, 'În siguranță', 9786069682371, 384, 'Thriller, Horror');

INSERT INTO Carte VALUES(10, 'Aventurile lui Tom Sawyer', 9789975545778, 260, 'Aventura');

INSERT INTO Carte VALUES(11, 'Robinson Crusoe', 9789975545747, 276, 'Aventura');

INSERT INTO Carte VALUES(12, 'Fundația', 9786069000724, 264, 'Science Fiction');

INSERT INTO Carte VALUES(13, 'Jocul lui Ender', 9786064311948, 352, 'Science Fiction');

INSERT INTO Carte VALUES(14, 'Metro 2035', 9786069000915, 608, 'Science Fiction');

INSERT INTO Carte VALUES(15, 'The Hobbit', 9780261102217, 285, 'Fantasy');

INSERT INTO Carte VALUES(16, 'Harry Potter și Camera Secretelor', 9786060861324, 400, 'Fantasy');

INSERT INTO Carte VALUES(17, 'Harry Potter și Piatra Filosofala', 9786060861317, 223, 'Fantasy');

INSERT INTO Carte VALUES(18, 'Harry Potter și Pocalul de Foc', 9786060862338, 721, 'Fantasy');

INSERT INTO Carte VALUES(19, 'Twilight', 9781904233657, 464, 'Romance');

INSERT INTO Carte VALUES(20, 'Monarhiile secolului XXI', 9786065885288, 324, 'Istorie');

INSERT INTO Carte VALUES(21, 'Palatul Elena Kretzulescu', 9786060810797, 184, 'Istorie');

INSERT INTO Carte VALUES(22, 'O fi bun de ceva?', 9786069801505, 448, 'Biografie');

INSERT INTO Carte VALUES(23, 'Jurnal', 9786069783726, 800, 'Biografie');

INSERT INTO Carte VALUES(24, 'Utopia', 9786068395920, 152, 'Filosofie');

Pentru tabelul AUTOR:

INSERT INTO Autor VALUES(10, 'Twain', 'Mark', '1910-04-21', 'S.U.A.');

INSERT INTO Autor VALUES(11, 'Christie', 'Agatha', '1890-09-15', 'Regatul Unit');

INSERT INTO Autor VALUES(12, 'Doyle', 'Arthur Conan', '1859-07-07', 'Regatul Unit');

INSERT INTO Autor VALUES(13, 'Swanson', 'Peter', '1968-05-26', 'S.U.A.');

INSERT INTO Autor VALUES(14, 'King', 'Stephen', '1947-09-21', 'S.U.A.');

INSERT INTO Autor VALUES(15, 'le Carre', 'John', '1931-10-19', 'Regatul Unit');

INSERT INTO Autor VALUES(16, 'Barnett', 'S.K.', null, null);

INSERT INTO Autor VALUES(17, 'Defoe', 'Daniel', null, 'Regatul Unit');

INSERT INTO Autor VALUES(18, 'Asimov', 'Isaac', '1920-01-02', 'Rusia');

INSERT INTO Autor VALUES(19, 'Card', 'Orson Scott', '1951-08-24', 'S.U.A.');

INSERT INTO Autor VALUES(20, 'Glukhovsky', 'Dmitry', '1979-06-12', 'Rusia');

INSERT INTO Autor VALUES(21, 'Tolkien', 'J. R. R.', '1973-09-02', 'Regatul Unit');

INSERT INTO Autor VALUES(22, 'Rowling', 'J. K.', '1965-07-31', 'Regatul Unit');



```
INSERT INTO Autor VALUES(23, 'Meyer', 'Stephenie', '1973-12-24', 'S.U.A.');
```

```
INSERT INTO Autor VALUES(24, 'Vlad', 'Ioan-Luca', null, 'România');
```

```
INSERT INTO Autor VALUES(25, 'Gâtejeanu-Gheorghe', 'Sandra', null, 'România');
```

```
INSERT INTO Autor VALUES(26, 'Mucenic', 'Cezara', null, 'România');
```

```
INSERT INTO Autor VALUES(27, 'Velescu', 'Oliver', null, 'România');
```

```
INSERT INTO Autor VALUES(28, 'Seinfeld', 'Jerry', '1954-04-29', 'S.U.A.');
```

```
INSERT INTO Autor VALUES(29, 'Kafka', 'Franz', '1883-06-03', 'Cehia');
```

```
INSERT INTO Autor VALUES(30, 'More', 'Thomas', '1478-02-07', 'Regatul Unit');
```

Pentru tabelul SCRIE:

```
INSERT INTO Scie VALUES(1, 11);
```

```
INSERT INTO Scie VALUES(2, 11);
```

```
INSERT INTO Scie VALUES(3, 12);
```

```
INSERT INTO Scie VALUES(4, 13);
```

```
INSERT INTO Scie VALUES(5, 14);
```

```
INSERT INTO Scie VALUES(6, 15);
```

```
INSERT INTO Scie VALUES(7, 14);
```

```
INSERT INTO Scie VALUES(8, 14);
```

```
INSERT INTO Scie VALUES(9, 16);
```

```
INSERT INTO Scie VALUES(10, 10);
```

```
INSERT INTO Scie VALUES(11, 17);
```

```
INSERT INTO Scie VALUES(12, 18);
```

```
INSERT INTO Scie VALUES(13, 19);
```

```

INSERT INTO Scie VALUES(14, 20);
INSERT INTO Scie VALUES(15, 21);
INSERT INTO Scie VALUES(16, 22);
INSERT INTO Scie VALUES(17, 22);
INSERT INTO Scie VALUES(18, 22);
INSERT INTO Scie VALUES(19, 23);
INSERT INTO Scie VALUES(20, 24);
INSERT INTO Scie VALUES(20, 25);
INSERT INTO Scie VALUES(21, 26);
INSERT INTO Scie VALUES(21, 27);
INSERT INTO Scie VALUES(22, 28);
INSERT INTO Scie VALUES(23, 29);
INSERT INTO Scie VALUES(24, 30);

```

Pentru tabelul CITIOR:

```

INSERT INTO Cititor VALUES(1, 'Traian', 'Apostol', '+40702032756');
INSERT INTO Cititor VALUES(2, 'Ardelean', 'Costache', '+40702015109');
INSERT INTO Cititor VALUES(3, 'Constantin', 'Remus', '+40702039282');
INSERT INTO Cititor VALUES(4, 'Albu', 'Dragoș', null);
INSERT INTO Cititor VALUES(5, 'Lupei', 'Teodora', null);
INSERT INTO Cititor VALUES(6, 'Radu', 'Carmen', '+40702005823');
INSERT INTO Cititor VALUES(7, 'Gheorghe', 'Clara', '+40710215288');
INSERT INTO Cititor VALUES(8, 'Petrescu', 'Casandra', null);

```

INSERT INTO Cititor VALUES(9, 'Albert', 'Iuliana', '+40702030837');

Pentru tabelul IMPRUMUT:

INSERT INTO Imprumut VALUES(12, 1);

INSERT INTO Imprumut VALUES(21, 2);

INSERT INTO Imprumut VALUES(14, 3);

INSERT INTO Imprumut VALUES(8, 4);

INSERT INTO Imprumut VALUES(12, 4);

INSERT INTO Imprumut VALUES(1, 5);

INSERT INTO Imprumut VALUES(24, 7);

INSERT INTO Imprumut VALUES(7, 7);

INSERT INTO Imprumut VALUES(16, 7);

INSERT INTO Imprumut VALUES(5, 8);

INSERT INTO Imprumut VALUES(2, 9);

INSERT INTO Imprumut VALUES(19, 9);

Pentru tabelul LOCATIE:

INSERT INTO Locatie VALUES(1, 'România', 'București', 'Sector 6', 'Str. Independenței', '319A');

INSERT INTO Locatie VALUES(2, 'România', 'București', 'Sector 1', 'Str. Iani Buzoiani', null);

INSERT INTO Locatie VALUES(3, 'Regatul Unit', 'Londra', null, 'Victoria Embankment', '50');

INSERT INTO Locatie VALUES(4, 'România', 'București', 'Sector 1', 'Str. Moeciu', '7A');

INSERT INTO Locatie VALUES(5, 'România', 'București', 'Sector 3', 'Bvd. Lacul Tei', '124');

INSERT INTO Locatie VALUES(6, 'România', 'București', 'Sector 1', 'Piața Presei Libere', '1');

Pentru tabelul EDITURA:

```
INSERT INTO Editura VALUES(10, 'Vellant', 1);
```

```
INSERT INTO Editura VALUES(11, 'Nemira', 2);
```

```
INSERT INTO Editura VALUES(12, 'Hodder & Stoughton', 3);
```

```
INSERT INTO Editura VALUES(13, 'Litera', 4);
```

```
INSERT INTO Editura VALUES(14, 'Humanitas', 6);
```

Pentru tabelul EDITIE:

```
INSERT INTO Editie VALUES(1, 10, 2004);
```

```
INSERT INTO Editie VALUES(1, 10, 2006);
```

```
INSERT INTO Editie VALUES(2, 10, 2010);
```

```
INSERT INTO Editie VALUES(3, 10, 2009);
```

```
INSERT INTO Editie VALUES(4, 10, 2002);
```

```
INSERT INTO Editie VALUES(5, 12, 2007);
```

```
INSERT INTO Editie VALUES(6, 11, 2016);
```

```
INSERT INTO Editie VALUES(6, 13, 2013);
```

```
INSERT INTO Editie VALUES(7, 11, 2012);
```

```
INSERT INTO Editie VALUES(8, 11, 2009);
```

```
INSERT INTO Editie VALUES(9, 11, 2020);
```

```
INSERT INTO Editie VALUES(10, 11, 2004);
```

```
INSERT INTO Editie VALUES(11, 11, 2018);
```

```
INSERT INTO Editie VALUES(12, 11, 2019);
```

```
INSERT INTO Editie VALUES(13, 11, 2017);
```

INSERT INTO Editie VALUES(14, 12, 2004);  
INSERT INTO Editie VALUES(15, 12, 2004);  
INSERT INTO Editie VALUES(16, 13, 2011);  
INSERT INTO Editie VALUES(16, 10, 2009);  
INSERT INTO Editie VALUES(16, 11, 2007);  
INSERT INTO Editie VALUES(17, 13, 2012);  
INSERT INTO Editie VALUES(18, 13, 2013);  
INSERT INTO Editie VALUES(19, 12, 2006);  
INSERT INTO Editie VALUES(20, 11, 2006);  
INSERT INTO Editie VALUES(21, 11, 2019);  
INSERT INTO Editie VALUES(22, 11, 2012);  
INSERT INTO Editie VALUES(23, 11, 2013);  
INSERT INTO Editie VALUES(24, 11, 2021);