

Assignment 2: Design a database schema for a library system, including tables, fields, and constraints like NOT NULL, UNIQUE, and CHECK. Include primary and foreign keys to establish relationships between tables.

Solution:

To design a database schema for a library system, we need to identify the main entities involved and their relationships. For this scenario, we can consider the following entities:

1. **Books**
2. **Authors**
3. **Members**
4. **Loans**
5. **Genres**

Let's define the tables, fields, and constraints:

Books Table

- **BookID** (Primary Key, INT, NOT NULL, AUTO_INCREMENT)
- **Title** (VARCHAR(255), NOT NULL)
- **ISBN** (VARCHAR(13), UNIQUE, NOT NULL)
- **Publisher** (VARCHAR(255), NOT NULL)
- **YearPublished** (YEAR, NOT NULL, CHECK (YearPublished >= 1450 AND YearPublished <= YEAR(CURDATE())))
- **GenreID** (Foreign Key, INT, NOT NULL)

Authors Table

- **AuthorID** (Primary Key, INT, NOT NULL, AUTO_INCREMENT)
- **FirstName** (VARCHAR(255), NOT NULL)
- **LastName** (VARCHAR(255), NOT NULL)

Books_Authors Table (Many-to-Many relationship between Books and Authors)

- **BookID** (Foreign Key, INT, NOT NULL)
- **AuthorID** (Foreign Key, INT, NOT NULL)
- **PRIMARY KEY (BookID, AuthorID)**

Members Table

- **MemberID** (Primary Key, INT, NOT NULL, AUTO_INCREMENT)
- **FirstName** (VARCHAR(255), NOT NULL)
- **LastName** (VARCHAR(255), NOT NULL)
- **Email** (VARCHAR(255), UNIQUE, NOT NULL)
- **Phone** (VARCHAR(15), NOT NULL)
- **Address** (VARCHAR(255), NOT NULL)
- **MembershipDate** (DATE, NOT NULL)

Loans Table

- **LoanID** (Primary Key, INT, NOT NULL, AUTO_INCREMENT)
- **BookID** (Foreign Key, INT, NOT NULL)
- **MemberID** (Foreign Key, INT, NOT NULL)
- **LoanDate** (DATE, NOT NULL)
- **ReturnDate** (DATE, NULL)
- **DueDate** (DATE, NOT NULL)
- **CHECK (ReturnDate IS NULL OR ReturnDate >= LoanDate)**

Genres Table

- **GenreID** (Primary Key, INT, NOT NULL, AUTO_INCREMENT)
- **GenreName** (VARCHAR(255), UNIQUE, NOT NULL)

Schema Definition

```
sql

CREATE TABLE Genres (
    GenreID INT AUTO_INCREMENT PRIMARY KEY,
    GenreName VARCHAR(255) UNIQUE NOT NULL
);

CREATE TABLE Books (
    BookID INT AUTO_INCREMENT PRIMARY KEY,
    Title VARCHAR(255) NOT NULL,
    ISBN VARCHAR(13) UNIQUE NOT NULL,
    Publisher VARCHAR(255) NOT NULL,
    YearPublished YEAR NOT NULL CHECK
    (YearPublished ≥ 1450 AND YearPublished ≤ YEAR(CURDATE())),
    GenreID INT NOT NULL,
    FOREIGN KEY (GenreID) REFERENCES Genres(GenreID)
);
```

```
sql

CREATE TABLE Authors (
    AuthorID INT AUTO_INCREMENT PRIMARY KEY,
    FirstName VARCHAR(255) NOT NULL,
    LastName VARCHAR(255) NOT NULL
);

CREATE TABLE Books_Authors (
    BookID INT NOT NULL,
    AuthorID INT NOT NULL,
    PRIMARY KEY (BookID, AuthorID),
    FOREIGN KEY (BookID) REFERENCES Books(BookID),
    FOREIGN KEY (AuthorID) REFERENCES Authors(AuthorID)
);
```

```
sql

CREATE TABLE Members (
    MemberID INT AUTO_INCREMENT PRIMARY KEY,
    FirstName VARCHAR(255) NOT NULL,
    LastName VARCHAR(255) NOT NULL,
    Email VARCHAR(255) UNIQUE NOT NULL,
    Phone VARCHAR(15) NOT NULL,
    Address VARCHAR(255) NOT NULL,
    MembershipDate DATE NOT NULL
);
```

```
sql

CREATE TABLE Loans (
    LoanID INT AUTO_INCREMENT PRIMARY KEY,
    BookID INT NOT NULL,
    MemberID INT NOT NULL,
    LoanDate DATE NOT NULL,
    ReturnDate DATE NULL,
    DueDate DATE NOT NULL,
    CHECK (ReturnDate IS NULL OR ReturnDate ≥ LoanDate),
    FOREIGN KEY (BookID) REFERENCES Books(BookID),
    FOREIGN KEY (MemberID) REFERENCES Members(MemberID)
);
```