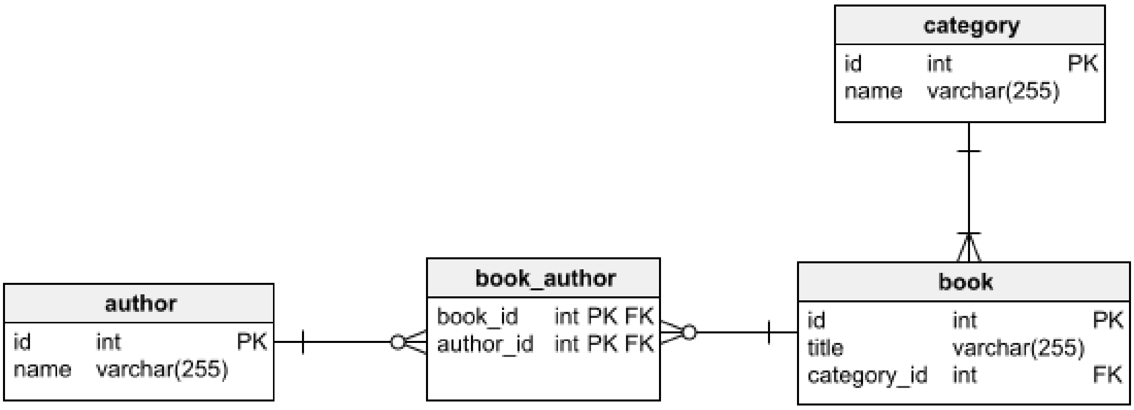
Assignment 02:

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.



Tables:

1. **Books**
   - Fields:
     - book_id (INT, PRIMARY KEY, NOT NULL, AUTO_INCREMENT): Unique identifier for the book.
     - title (VARCHAR(255), NOT NULL): Title of the book.
     - isbn (VARCHAR(13), UNIQUE, CHECK (LENGTH(isbn) = 13)): International

        Standard Book Number (ISBN) for the book (enforces 13 characters).
        - publication_year (INT, NOT NULL): Year the book was published.
        - edition (INT): Edition number of the book (optional).
    - Constraints:
        - PRIMARY KEY (book_id): Ensures each book has a unique identifier.
        - UNIQUE (isbn): Ensures no duplicate ISBNs exist.
        - CHECK (LENGTH(isbn) = 13): Ensures the ISBN has the correct length (13 characters).

2. **Authors**
    - Fields:
        - author_id (INT, PRIMARY KEY, NOT NULL, AUTO_INCREMENT): Unique identifier for the author.
        - first_name (VARCHAR(50), NOT NULL): Author's first name.
        - last_name (VARCHAR(50), NOT NULL): Author's last name.

- Constraints:
  - PRIMARY KEY (author_id): Ensures each author has a unique identifier.

3. **Book_Authors (Many-to-Many Relationship between Books and Authors)**
   - Fields:
     - book_id (INT, FOREIGN KEY REFERENCES Books(book_id), NOT NULL): References the book_id in the Books table.
     - author_id (INT, FOREIGN KEY REFERENCES Authors(author_id), NOT NULL): References the author_id in the Authors table.
   - Constraints:
     - PRIMARY KEY (book_id, author_id): Composite primary key to ensure a unique relationship between a book and an author.
     - FOREIGN KEY (book_id) REFERENCES Books(book_id): Foreign key referencing the Books table.

        - FOREIGN KEY (author_id) REFERENCES Authors(author_id): Foreign key referencing the Authors table.

4. **Customers**
    - Fields:
        - customer_id (INT, PRIMARY KEY, NOT NULL, AUTO_INCREMENT): Unique identifier for the customer.
        - name (VARCHAR(255), NOT NULL): Customer's name.
        - email (VARCHAR(255), UNIQUE, NOT NULL): Customer's email address (enforces uniqueness).
    - Constraints:
        - PRIMARY KEY (customer_id): Ensures each customer has a unique identifier.
        - UNIQUE (email): Ensures no duplicate email addresses exist.
5. **Loans**
    - Fields:
        - loan_id (INT, PRIMARY KEY, NOT NULL, AUTO_INCREMENT): Unique identifier for the loan.

- book_id (INT, FOREIGN KEY REFERENCES Books(book_id), NOT NULL): References the book_id in the Books table.
- customer_id (INT, FOREIGN KEY REFERENCES Customers(customer_id), NOT NULL): References the customer_id in the Customers table.
- loan_date (DATE, NOT NULL): Date the book was loaned.
- due_date (DATE, NOT NULL): Date the book is due for return.
- return_date (DATE): Date the book was returned (optional).

- Constraints:
    - PRIMARY KEY (loan_id): Ensures each loan has a unique identifier.
    - FOREIGN KEY (book_id) REFERENCES Books(book_id): Foreign key referencing the Books table.
    - FOREIGN KEY (customer_id) REFERENCES Customers(customer_id): Foreign key referencing the Customers table.