**Q: Library management system - Design a class to manage library resources, including books, journals, and magazines, borrowing, and returning books.**

**Approach to solve:**

1.Understand the problem statement clearly and find out the all the end points need to be created .

2.Figure out the proper datasturctures to manipulate the data in the api

3.Figure out the database design with proper list of tables and appropriate constraints to solve the problem.

**Pseudocode:**

**GET\_BOOKS (HTTP GET method):**

Step 1: Create a database connection using the set\_connection() method.

Step 2: Execute a SELECT query to retrieve all books from the books table.

Step 3: Fetch all the rows and convert the rows into a list of dictionaries.

Step 4: Close all the connections.

Step 5: Return the list of books as a JSON object along with the status code 200.

**ADD\_BOOK (HTTP POST method):**

Step 1: Extract the book data from the JSON payload.

Step 2: If the data format is incorrect or any required field is missing, display a message saying "Missing required field(s)".

Step 3: Create a database connection using the set\_connection() method. If it is created successfully, execute the INSERT query to insert the book data into the books table.

Step 4: Commit the transaction and close all the connections.

Step 5: Return a message saying "Book added successfully" along with the status code 201.

**BORROW\_BOOK (HTTP PUT method):**

Step 1: Extract the book ID from the JSON payload.

Step 2: Create a database connection using the set\_connection() method.

Step 3: Check if the book is available by executing a SELECT query.

Step 4: If the book is not available, return a message saying "Book is not available" along with the status code 400.

Step 5: If the book is available, update the book's availability to "false" by executing an UPDATE query.

Step 6: Commit the transaction and close all the connections.

Step 7: Return a message saying "Book borrowed successfully" along with the status code 200.

**RETURN\_BOOK (HTTP PUT method):**

Step 1: Extract the book ID from the JSON payload.

Step 2: Create a database connection using the set\_connection() method.

Step 3: Check if the book is borrowed by executing a SELECT query.

Step 4: If the book is not borrowed, return a message saying "Book is not borrowed" along with the status code 400.

Step 5: If the book is borrowed, update the book's availability to "true" by executing an UPDATE query.

Step 6: Commit the transaction and close all the connections.

Step 7: Return a message saying "Book returned successfully" along with the status code 200.