Project-based practice assignment on MySQL:

Name : Rohan Jain T B

Project: Online Library Management System

Description: In this project, you will create a database for an online library management system that will keep track of books, authors, and users. The system will allow users to borrow and return books, and the librarian to manage the inventory of the library.

Requirements:
Create a database named 'library'.
Create tables for books, authors, and users.

The 'books' table should have the following columns:
book_id (integer, primary key)
title (varchar)
author_id (integer, foreign key)
publisher (varchar)
publish_date (date)
quantity (integer)

The 'authors' table should have the following columns:
author_id (integer, primary key)
first_name (varchar)
last_name (varchar)
email (varchar)

The 'users' table should have the following columns:
user_id (integer, primary key)
first_name (varchar)
last_name (varchar)
email (varchar)
password (varchar)

Create a table for borrowed books.

The 'borrowed_books' table should have the following columns:
id (integer, primary key)
book_id (integer, foreign key)
user_id (integer, foreign key)
borrowed_date (date)
due_date (date)
returned_date (date)

Insert some sample data into the tables.

Write SQL queries to perform the following operations:

1. Display all the books in the library.
   select * from books;
2. Display all the authors in the library.
   select * from authors;
3. Display all the users in the library.
   select * from users;
4. Display all the borrowed books.
   select * from borrowed_books;
5. Display all the books borrowed by a particular user.

```
SELECT books.title, books.author_id, books.publisher,
books.publish_date, borrowed_books.borrowed_date,
borrowed_books.due_date, borrowed_books.returned_date
FROM borrowed_books
JOIN books ON borrowed_books.book_id = books.book_id
WHERE borrowed_books.user_id =user_id;
```

6. Display all the books written by a particular author.

```
select book_id,title,publisher from books join authors on
books.author_id=authors.author_id ;
```

7. Display the number of available copies for a particular book.

```
SELECT books.title, books.quantity - COUNT(book_id) AS
available_copies
FROM books group by book_id;
```

8. Add a new book to the library.

```
INSERT INTO books (book_id,title,author_id ,publisher, publish_date,
quantity)
VALUES (6,'Jungel Book',author_id, 'New Publisher', '2020-03-21', 10);
```

9. Update the quantity of a book in the library.

```
UPDATE books SET quantity = 8 WHERE book_id = 2;
```

10. Delete a book from the library.

```
DELETE FROM books WHERE  author_id=3;
```

11. Add a new user to the library.

```
INSERT INTO users (user_id,first_name, last_name, email, password)
VALUES (4,'John', 'Doe', 'johndoe@example.com', 'password123');
```

12. Update the password of a user.

```
UPDATE users SET password = 'Rohan@123' WHERE user_id=3;
```

13. Delete a user from the library.

```
DELETE FROM users WHERE password = 'password1234';
```

14. Borrow a book.

```
UPDATE borrowed_books SET  book_id = user_id, borrowed_date =
'2022-01-02', due_date = '2022-01-16' WHERE id = 3 ;
```

15. Return a book.

```
UPDATE books
SET quantity = quantity + 1
WHERE book_id = 1;
```

Test your queries to ensure they are working properly.

Create a web interface for the library management system using ReactJS/Node JS and MySQL.