

## Task 7: Review and Test the Object-Oriented Programming Solution

### Objective

The objective is to review and test the Library Management System application by analyzing actual test results against expected results to identify discrepancies.

### Test Cases

1. **Test Case 1: User Borrows an Available Book**
  - **Input:** User "Alice Smith" borrows "The Great Gatsby".
  - **Expected Result:** Book status changes to "Not Available", and the book is added to Alice's borrowed books list.
  - **Actual Result:** [To be filled after testing]
2. **Test Case 2: User Tries to Borrow an Unavailable Book**
  - **Input:** User "Bob Johnson" borrows "The Great Gatsby" after Alice has already borrowed it.
  - **Expected Result:** Operation fails with a message indicating the book is not available.
  - **Actual Result:** [To be filled after testing]
3. **Test Case 3: User Returns a Book**
  - **Input:** User "Alice Smith" returns "The Great Gatsby".
  - **Expected Result:** Book status changes to "Available", and the book is removed from Alice's borrowed books list.
  - **Actual Result:** [To be filled after testing]
4. **Test Case 4: Search for a Book by Title**
  - **Input:** Search for books with title containing "1984".
  - **Expected Result:** List containing "1984" by George Orwell.
  - **Actual Result:** [To be filled after testing]
5. **Test Case 5: Register a New User**
  - **Input:** Register a new user "Charlie Brown".
  - **Expected Result:** User is added to the library's user list.
  - **Actual Result:** [To be filled after testing]
6. **Test Case 6: Error Handling for Invalid Book ID**
  - **Input:** User tries to borrow a book with an invalid ID "999".
  - **Expected Result:** Operation fails with a message indicating the book is not found.
  - **Actual Result:** [To be filled after testing]

### Testing Code

Here is the updated test code to include these tests and capture the actual results:

See File library\_test.py

### Analyzing Test Results

1. **Test Case 1: User Borrows an Available Book**
  - **Expected Result:** Book status changes to "Not Available", and the book is added to Alice's borrowed books list.

- **Actual Result:** [Results from running the code]
- 2. **Test Case 2: User Tries to Borrow an Unavailable Book**
  - **Expected Result:** Operation fails with a message indicating the book is not available.
  - **Actual Result:** [Results from running the code]
- 3. **Test Case 3: User Returns a Book**
  - **Expected Result:** Book status changes to "Available", and the book is removed from Alice's borrowed books list.
  - **Actual Result:** [Results from running the code]
- 4. **Test Case 4: Search for a Book by Title**
  - **Expected Result:** List containing "1984" by George Orwell.
  - **Actual Result:** [Results from running the code]
- 5. **Test Case 5: Register a New User**
  - **Expected Result:** User is added to the library's user list.
  - **Actual Result:** [Results from running the code]
- 6. **Test Case 6: Error Handling for Invalid Book ID**
  - **Expected Result:** Operation fails with a message indicating the book is not found.
  - **Actual Result:** [Results from running the code]

## Conclusion

After running the test cases, analyse the actual results and compare them with the expected results to identify any discrepancies. Any discrepancies identified will need to be addressed by reviewing and possibly revising the code to ensure it meets the expected functionality.

## **Recommendations**

1. **Enhance Testing Coverage:**
  - Continue to add more test cases to cover different scenarios and edge cases.
2. **Improve Error Handling:**
  - Ensure robust error handling throughout the application to gracefully handle invalid inputs and operations.
3. **Optimize Performance:**
  - Review the performance, especially for search operations, and consider optimisations as needed.
4. **User Experience:**
  - Implement user-friendly features and improve the overall usability based on feedback.

By following these steps, the Library Management System can be improved and refined to meet user expectations and function reliably.