Library Management System - Project Documentation

1. What is this project?

This is a console-based Library Management System developed in Java using MySQL. It helps manage books, members, and borrowing operations. It features CRUD operations and uses JDBC for connecting Java to MySQL.

2. Why this project?

It reduces manual work in libraries and is a great way to learn Java, JDBC, and MySQL database management. It automates book availability, keeps records safe, and teaches database-driven application development.

3. Technologies Used

- Java (JDK 8+)
- JDBC (MySQL Connector/J 9.3.0)
- MySQL (Database)
- VS Code or IntelliJ (IDE)
- Git (Version control)

4. JDBC Connection Process

1. Load Driver: Class.forName("com.mysql.cj.jdbc.Driver")
2. Establish Connection: DriverManager.getConnection(DB_URL, USER, PASS)
3. Use PreparedStatement to perform SQL operations
4. Handle results using ResultSet

5. Java Package Structure

LibraryManagement/

lib/

mysql-connector-j-9.3.0.jar

```
src/
   db/  DBConnection.java
   manager/  LibraryManager.java
Main.java
```

6. Purpose of Each Class

- DBConnection.java: Handles DB connection
- LibraryManager.java: Business logic (CRUD, issue/return, search)
- Main.java: User menu and interaction

7. MySQL Tables

- books(id, title, author, isbn, category, available)
- members(id, name)
- borrow_history(id, member_id, book_id, borrow_date, return_date)

8. Sample Menu

1. Add Book
2. Register Member
3. View All Books
4. Issue Book
5. Return Book
6. Delete Book
7. Delete Member
8. Exit
9. Search Book by Title

9. Why Console-Based?

Easier to focus on logic, quick to test and demonstrate, best for JDBC learning and understanding.

10. Common Viva/Interview Questions

Q1: What is JDBC?

A: JDBC is Java API to connect to databases using SQL.

Q2: Why use PreparedStatement?

A: It prevents SQL injection and handles dynamic queries.

Q3: What happens when a book is issued?

A: Entry added to borrow_history, book marked unavailable.

Q4: How is return handled?

A: return_date is set and book availability updated.

Q5: What if book/member has existing history?

A: We delete the history first (foreign key constraint).

Q6: Why use layered architecture?

A: To separate concerns UI, logic, DB connection.

Q7: Can this project be extended?

A: Yes, by adding a web UI or mobile frontend.

11. Project Architecture Diagram

Library Management System - Project Documentation