**METHODS FOR FUNCTIONALITY :**

***Management\_Sys.java :***

1. **main :** This is where the main menu page is located along with essential code to call all the linked pages
2. **add\_student , delete\_student, update\_student :** Method for adding , deleting and updating student profiles which are created from the Student.java file
3. **add\_book , delete\_book , update\_book :** Method for adding , deleting and updating books which are stored in the book.java file
4. **student\_management :** It consists of the student management page which requires admin access (through password), it contains functionality for adding updating and deleting student profiles, which are arranged for easy access
5. **book\_management :** It consists of the book management page which requires admin access (through password), it contains functionality for adding updating and deleting books, which are arranged in a page for easy access
6. **Library() :** This method provides easy access for all students to issue, return books and generate fine reports

***Student.java :***

1. **Student() (constructor) :** This is the constructor for the student profile which stores all the information like name , age etc in the profile
2. **Update() :** For updating the student profile
3. **Issue\_book :** for issueing a book from the prespective of the student
4. **return\_book :** To return the books already issued
5. **Calcualte\_fine :** To generate a fine report if the time is beyond the issue limit for the books a student has issued

***Book.java :***

This file consists of just the ArrayList and HashMaps for the books and their repective issue limits

**CONCLUSION :** All these methods defined here contruct a basic library management system that allows both advanced admin functionality and simple student interfaces for basic operations like issuing and returning books etc