GITHUB CODE

#include <iostream>

#include <vector>

#include <string>

using namespace std;

class Book {

public:

int id;

string title;

string author;

bool isIssued;

Book(int i, string t, string a) : id(i), title(t), author(a), isIssued(false) {}

};

class Library {

private:

vector<Book> books;

public:

void addBook() {

int id;

string title, author;

cout << "Enter Book ID: ";

cin >> id;

cin.ignore(); // To handle newline character after entering ID

cout << "Enter Book Title: ";

getline(cin, title);

cout << "Enter Book Author: ";

getline(cin, author);

books.push\_back(Book(id, title, author));

cout << "Book added successfully!" << endl;

}

void displayBooks() {

if (books.empty()) {

cout << "No books available in the library." << endl;

} else {

cout << "Books available in the library:" << endl;

for (const auto &book : books) {

cout << "ID: " << book.id << ", Title: " << book.title << ", Author: " << book.author

<< ", Status: " << (book.isIssued ? "Issued" : "Available") << endl;

}

}

}

void issueBook() {

int id;

cout << "Enter Book ID to issue: ";

cin >> id;

for (auto &book : books) {

if (book.id == id) {

if (!book.isIssued) {

book.isIssued = true;

cout << "Book issued successfully!" << endl;

} else {

cout << "Book is already issued." << endl;

}

return;

}

}

cout << "Book with ID " << id << " not found." << endl;

}

void returnBook() {

int id;

cout << "Enter Book ID to return: ";

cin >> id;

for (auto &book : books) {

if (book.id == id) {

if (book.isIssued) {

book.isIssued = false;

cout << "Book returned successfully!" << endl;

} else {

cout << "This book was not issued." << endl;

}

return;

}

}

cout << "Book with ID " << id << " not found." << endl;

}

};

int main() {

Library lib;

int choice;

while (true) {

cout << "\nLibrary Management System\n";

cout << "1. Add Book\n";

cout << "2. Display Books\n";

cout << "3. Issue Book\n";

cout << "4. Return Book\n";

cout << "5. Exit\n";

cout << "Enter your choice: ";

cin >> choice;

switch (choice) {

case 1:

lib.addBook();

break;

case 2:

lib.displayBooks();

break;

case 3:

lib.issueBook();

break;

case 4:

lib.returnBook();

break;

case 5:

cout << "Exiting..." << endl;

return 0;

default:

cout << "Invalid choice. Please try again." << endl;

       }

    }

}