**M.YUVARAJ**

**ᴄ++ ᴘʀᴏᴊᴇᴄᴛ**

**Design a program to manage a library’s inventory, allowing users to add, remove, and search for books, while also keeping track of borrowed and returned books.**

**AIM:  
 Design a program to manage a library’s inventory, allowing users to add, remove, and search for books, while also keeping track of borrowed and returned books.**

**PROGRAM:**

**#include <iostream>**

**using namespace std;**

**const int MAX\_BOOKS = 100; // Maximum number of books in the library**

**// Structure to represent a book**

**struct Book {**

**string title;**

**string author;**

**bool available; // Flag to check if the book is available for borrowing**

**};**

**// Function prototypes**

**void addBook(Book library[], int &numBooks);**

**void removeBook(Book library[], int &numBooks);**

**void searchBook(const Book library[], int numBooks);**

**void displayInventory(const Book library[], int numBooks);**

**void borrowBook(Book library[], int numBooks);**

**void returnBook(Book library[], int numBooks);**

**int main() {**

**Book library[MAX\_BOOKS]; // Array to store the library's inventory**

**int numBooks = 0; // Number of books in the library**

**int choice;**

**do {**

**// Display menu**

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

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

**cout << "2. Remove Book\n";**

**cout << "3. Search Book\n";**

**cout << "4. Display Inventory\n";**

**cout << "5. Borrow Book\n";**

**cout << "6. Return Book\n";**

**cout << "0. Exit\n";**

**cout << "Enter your choice: ";**

**cin >> choice;**

**// Process user's choice**

**switch (choice) {**

**case 1:**

**addBook(library, numBooks);**

**break;**

**case 2:**

**removeBook(library, numBooks);**

**break;**

**case 3:**

**searchBook(library, numBooks);**

**break;**

**case 4:**

**displayInventory(library, numBooks);**

**break;**

**case 5:**

**borrowBook(library, numBooks);**

**break;**

**case 6:**

**returnBook(library, numBooks);**

**break;**

**case 0:**

**cout << "Exiting the program. \n";**

**break;**

**default:**

**cout << "Invalid choice. Please enter a valid option.\n";**

**}**

**} while (choice != 0);**

**return 0;**

**}**

**// Function to add a book to the library**

**void addBook(Book library[], int &numBooks) {**

**if (numBooks < MAX\_BOOKS) {**

**Book newBook;**

**cout << "Enter book title: ";**

**cin.ignore(); // Ignore the newline character in the input buffer**

**getline(cin, newBook.title);**

**cout << "Enter author: ";**

**getline(cin, newBook.author);**

**newBook.available = true;**

**library[numBooks++] = newBook;**

**cout << "Book added successfully!\n";**

**} else {**

**cout << "The library is full. Cannot add more books.\n";**

**}**

**}**

**// Function to remove a book from the library**

**void removeBook(Book library[], int &numBooks) {**

**string titleToRemove;**

**cout << "Enter the title of the book to remove: ";**

**cin.ignore();**

**getline(cin, titleToRemove);**

**for (int i = 0; i < numBooks; ++i) {**

**if (library[i].title == titleToRemove) {**

**for (int j = i; j < numBooks - 1; ++j) {**

**library[j] = library[j + 1];**

**}**

**numBooks--;**

**cout << "Book removed successfully!\n";**

**return;**

**}**

**}**

**cout << "Book not found in the library.\n";**

**}**

**// Function to search for a book in the library**

**void searchBook(const Book library[], int numBooks) {**

**string searchTitle;**

**cout << "Enter the title of the book to search: ";**

**cin.ignore();**

**getline(cin, searchTitle);**

**for (int i = 0; i < numBooks; ++i) {**

**if (library[i].title == searchTitle) {**

**cout << "Book found!\n";**

**cout << "Title: " << library[i].title << "\n";**

**cout << "Author: " << library[i].author << "\n";**

**cout << "Availability: " << (library[i].available ? "Available" : "Not Available") << "\n";**

**return;**

**}**

**}**

**cout << "Book not found in the library.\n";**

**}**

**// Function to display the library's inventory**

**void displayInventory(const Book library[], int numBooks) {**

**if (numBooks == 0) {**

**cout << "The library is empty.\n";**

**} else {**

**cout << "Library Inventory:\n";**

**for (int i = 0; i < numBooks; ++i) {**

**cout << "Title: " << library[i].title << "\n";**

**cout << "Author: " << library[i].author << "\n";**

**cout << "Availability: " << (library[i].available ? "Available" : "Not Available") << "\n";**

**cout << "------------------------\n";**

**}**

**}**

**}**

**// Function to borrow a book from the library**

**void borrowBook(Book library[], int numBooks) {**

**string borrowTitle;**

**cout << "Enter the title of the book to borrow: ";**

**cin.ignore();**

**getline(cin, borrowTitle);**

**for (int i = 0; i < numBooks; ++i) {**

**if (library[i].title == borrowTitle) {**

**if (library[i].available) {**

**library[i].available = false;**

**cout << "Book borrowed successfully!\n";**

**} else {**

**cout << "The book is not available for borrowing.\n";**

**}**

**return;**

**}**

**}**

**cout << "Book not found in the library.\n";**

**}**

**// Function to return a book to the library**

**void returnBook(Book library[], int numBooks) {**

**string returnTitle;**

**cout << "Enter the title of the book to return: ";**

**cin.ignore();**

**getline(cin, returnTitle);**

**for (int i = 0; i < numBooks; ++i) {**

**if (library[i].title == returnTitle) {**

**if (!library[i].available) {**

**library[i].available = true;**

**cout << "Book returned successfully!\n";**

**} else {**

**cout << "This book is already available in the library.\n";**

**}**

**return;**

**}**

**}**

**cout << "Book not found in the library.\n";**

**}  
  
  
OUTPUT:**

**Library Inventory Management System**

**1. Add Book**

**2. Remove Book**

**3. Search Book**

**4. Display Inventory**

**5. Borrow Book**

**6. Return Book**

**0. Exit**

**Enter your choice: 1**

**Enter book title: c++**

**Enter author: yuvaraj**

**Book added successfully!**

**Library Inventory Management System**

**1. Add Book**

**2. Remove Book**

**3. Search Book**

**4. Display Inventory**

**5. Borrow Book**

**6. Return Book**

**0. Exit**

**Enter your choice: 1**

**Enter book title: python**

**Enter author: vidhun**

**Book added successfully!**

**Library Inventory Management System**

**1. Add Book**

**2. Remove Book**

**3. Search Book**

**4. Display Inventory**

**5. Borrow Book**

**6. Return Book**

**0. Exit**

**Enter your choice: 1**

**Enter book title: adp**

**Enter author: mohith**

**Book added successfully!**

**Library Inventory Management System**

**1. Add Book**

**2. Remove Book**

**3. Search Book**

**4. Display Inventory**

**5. Borrow Book**

**6. Return Book**

**0. Exit**

**Enter your choice: 1**

**Enter book title: ai**

**Enter author: samuel**

**Book added successfully!**

**Library Inventory Management System**

**1. Add Book**

**2. Remove Book**

**3. Search Book**

**4. Display Inventory**

**5. Borrow Book**

**6. Return Book**

**0. Exit**

**Enter your choice: 2**

**Enter the title of the book to remove: ai**

**Book removed successfully!**

**Library Inventory Management System**

**1. Add Book**

**2. Remove Book**

**3. Search Book**

**4. Display Inventory**

**5. Borrow Book**

**6. Return Book**

**0. Exit**

**Enter your choice: 4**

**Library Inventory:**

**Title: c++**

**Author: yuvaraj**

**Availability: Available**

**------------------------**

**Title: python**

**Author: vidhun**

**Availability: Available**

**------------------------**

**Title: adp**

**Author: mohith**

**Availability: Available**

**------------------------**

**Library Inventory Management System**

**1. Add Book**

**2. Remove Book**

**3. Search Book**

**4. Display Inventory**

**5. Borrow Book**

**6. Return Book**

**0. Exit**

**Enter your choice: 1**

**Enter book title: ai**

**Enter author: samuel**

**Book added successfully!**

**Library Inventory Management System**

**1. Add Book**

**2. Remove Book**

**3. Search Book**

**4. Display Inventory**

**5. Borrow Book**

**6. Return Book**

**0. Exit**

**Enter your choice: 3**

**Enter the title of the book to search: c++**

**Book found!**

**Title: c++**

**Author: yuvaraj**

**Availability: Available**

**Library Inventory Management System**

**1. Add Book**

**2. Remove Book**

**3. Search Book**

**4. Display Inventory**

**5. Borrow Book**

**6. Return Book**

**0. Exit**

**Enter your choice: 5**

**Enter the title of the book to borrow: c++**

**Book borrowed successfully!**

**Library Inventory Management System**

**1. Add Book**

**2. Remove Book**

**3. Search Book**

**4. Display Inventory**

**5. Borrow Book**

**6. Return Book**

**0. Exit**

**Enter your choice: 4**

**Library Inventory:**

**Title: c++**

**Author: yuvaraj**

**Availability: Not Available**

**------------------------**

**Title: python**

**Author: vidhun**

**Availability: Available**

**------------------------**

**Title: adp**

**Author: mohith**

**Availability: Available**

**------------------------**

**Title: ai**

**Author: samuel**

**Availability: Available**

**------------------------**

**Library Inventory Management System**

**1. Add Book**

**2. Remove Book**

**3. Search Book**

**4. Display Inventory**

**5. Borrow Book**

**6. Return Book**

**0. Exit**

**Enter your choice: 6**

**Enter the title of the book to return: c++**

**Book returned successfully!**

**Library Inventory Management System**

**1. Add Book**

**2. Remove Book**

**3. Search Book**

**4. Display Inventory**

**5. Borrow Book**

**6. Return Book**

**0. Exit**

**Enter your choice: 4**

**Library Inventory:**

**Title: c++**

**Author: yuvaraj**

**Availability: Available**

**------------------------**

**Title: python**

**Author: vidhun**

**Availability: Available**

**------------------------**

**Title: adp**

**Author: mohith**

**Availability: Available**

**------------------------**

**Title: ai**

**Author: samuel**

**Availability: Available**

**------------------------**

**Library Inventory Management System**

**1. Add Book**

**2. Remove Book**

**3. Search Book**

**4. Display Inventory**

**5. Borrow Book**

**6. Return Book**

**0. Exit**

**Enter your choice: 0**

**Exiting the program.   
  
  
DESCRIPTION:**

**The provided C++ code implements a simple Library Inventory Management System using a menu-driven console interface. It allows users to add, remove, search, display, borrow, and return books in the library. The program utilizes a structure named "Book" to store book information, including title, author, and availability status. The main function runs a loop to continuously prompt the user for actions until choosing to exit (option 0). The code employs functions for each operation, ensuring modular and organized code structure.**