



Library Management System

Youssef Taha / 241015238

A Brief description of the project:

- This project is a Library Management System that helps to manage books in a library. It provides simple tools to:
 - 1) Keep track of all the books in the library (inventory).
 - 2) Add new books or remove old ones from the library.
 - 3) Borrow books and return them, while updating their status.
 - 4) Search for a book by its title and see if it's available.
 - 5) Check if a borrowed book is overdue (borrowed for more than 14 days).
- The system is easy to use, with a menu that allows users to pick what they want to do. It also uses basic programming concepts like organizing information in tables, working with dates, and breaking tasks into smaller functions.

Basic Functions Implemented:

- 1) Initialize Inventory: Sets up a predefined list of books in the inventory for testing or initial use.
- 2) Show Menu: Displays the list of available actions (menu options) for the library management system.
- 3) View Inventory: Displays a list of all books in the inventory, showing details such as the book ID, title, author, department, and borrowing status.
- 4) Add Book: Allows the user to add a new book to the inventory by providing its title, author, department, and ID.
- 5) Delete Book: Enables the user to remove a book from the inventory by entering its title.
- 6) Borrow Book: Marks a book as borrowed in the inventory.
- 7) Return Book: Marks a book as returned in the inventory.
- 8) Search Book: Checks if a specific book is available in the inventory by title.
- 9) Overdue Books: Checks if a book is overdue based on its borrowing date.
- 10) Main Function: Serves as the entry point for the program and handles user interaction with the system.

The implemented code:

1) The Libraries:

`#include <stdio.h>`: Includes the standard input-output library

`#include <string.h>`: Includes the string library to use functions like (strcpy, strcmp and strcspn)

`#include <time.h>`: Includes the time library to work with date and time functionalities

```
#include <stdio.h>
#include <string.h>
#include <time.h>
```

2) The Struct that identifies the variables:

Defines a structure named (Book) that represents a book with attributes: (title, author, department).

(Id): A unique identifier for each book.

(Is_borrowed) : To indicate whether the book is borrowed (1) or not (0).

```
// A Struct to Define the variables of each book:
struct Book {
    char title[50];
    char author[50];
    char department[30];
    int id;
    int is_borrowed; // 0 = not borrowed, 1 = borrowed
};
```

3) Setting a global array to make the books inventory:

```
// Inventory of books (Global Array):
#define MAX_BOOKS 100
struct Book inventory[MAX_BOOKS];
int book_count = 0;
```

#define MAX_BOOKS 100 : Sets the maximum number of books the inventory can hold to 100 books.

struct Book inventory[MAX_BOOKS]; : Declares a global array inventory to store up to one hundred books.

int book_count = 0; : Keeps track of the current number of books in the inventory.

4) The Functions:

• Initialize Inventory:

- In this function I used the struct that I made to put some books in the inventory.
- I used strcpy because it copies the string into the specific place I want to put it in.

```
// Making the Inventory:
void initialize_inventory() {
    strcpy(inventory[0].title, "Calculus 1");
    strcpy(inventory[0].author, "Youssef Ahmed");
    strcpy(inventory[0].department, "Math");
    inventory[0].id = 1;
    inventory[0].is_borrowed = 0;

    strcpy(inventory[1].title, "Computing");
    strcpy(inventory[1].author, "Ahmed Mohamed");
    strcpy(inventory[1].department, "Computer Science");
    inventory[1].id = 2;
    inventory[1].is_borrowed = 0;

    strcpy(inventory[2].title, "Calculus 2");
    strcpy(inventory[2].author, "Ahmed Zaki");
    strcpy(inventory[2].department, "Math");
    inventory[2].id = 3;
    inventory[2].is_borrowed = 0;
}
```

• Show Menu:

- In this function I printed all the options to choose from.

```
// The Menu of the functions to choose from:
void show_menu() {
    printf("\nLibrary Management System\n");
    printf("1. View Books Inventory\n");
    printf("2. Add Book\n");
    printf("3. Delete Book\n");
    printf("4. Borrow Book\n");
    printf("5. Return Book\n");
    printf("6. Search Book\n");
    printf("7. View Overdue Books\n");
    printf("8. Exit\n");
}
```

- View Inventory:

- In this function I used a for loop to show all the books in the inventory.
- It runs from zero to the max book count I putted.
- I used the number of the struct as the variable (i).

```
// Showing the inventory function:
void view_inventory() {
    printf("\n--- Inventory ---\n");
    for (int i = 0; i < book_count; i++) {
        printf("ID: %d, Title: %s, Author: %s, Department: %s, Borrowed: %s\n",
            inventory[i].id, inventory[i].title, inventory[i].author,
            inventory[i].department,
            inventory[i].is_borrowed ? "Yes" : "No");
    }
}
```

- Add Book:

```
// Add book function:
void add_book() {
    if (book_count >= MAX_BOOKS) {
        printf("Inventory is full. Cannot add more books.\n");
        return;
    }

    struct Book new_book;
    printf("\n--- Add Book ---\n");
    printf("Enter the title of the book: ");
    fgets(new_book.title, sizeof(new_book.title), stdin);
    new_book.title[strcspn(new_book.title, "\n")] = 0; // Remove newline that fgets does
    printf("Enter the author of the book: ");
    fgets(new_book.author, sizeof(new_book.author), stdin);
    new_book.author[strcspn(new_book.author, "\n")] = 0; // Remove newline that fgets does
    printf("Enter the department of the book: ");
    fgets(new_book.department, sizeof(new_book.department), stdin);
    new_book.department[strcspn(new_book.department, "\n")] = 0; // Remove newline that fgets does
    printf("Enter the ID of the book: ");
    scanf("%d", &new_book.id);
    getchar();

    new_book.is_borrowed = 0; // Book will be added as not borrowed
    inventory[book_count++] = new_book; // Add to inventory

    printf("Book '%s' by '%s' added to the '%s' department with ID %d.\n",
        new_book.title, new_book.author, new_book.department, new_book.id);
}
```

- I used if condition to make sure a new book can't be added if the book count is at max (100 books).
- I made a variable called new_book of type struct book to store the details of the new book that will be added.
- I used (fgets) to read the string of the book from the user input and also to make sure that the doesn't exceed the size of the array.
- The line after each fgets to remove the newline that fgets does.
- I used getchar because without it the next fgets call would read the leftover newline instead of waiting for the user input.
- The new book will be added as not borrowed and book count will be increased by one.

• Delete Book:

- I used (fgets) to read the book title from the user.
- I used a for loop to search through the inventory for the book title.
- I used the if statement with a condition that compares the title input by the user with the title of each book in the inventory
- If a match is found then the condition is true and another loop will be initiated to shift the books after the deleted book up by one position.
- Book count will be decreased by one.
- I used return to exit the function after successfully deleting the book.
- If a match isn't found then a message will appear that a book with this title isn't found in the library.

```
// Delete book function:
void delete_books() {
    char title[50];
    printf("\n--- Delete Book ---\n");
    printf("Enter the title of the book to delete: ");
    fgets(title, sizeof(title), stdin);
    title[strcspn(title, "\n")] = 0; // Remove newline character

    for (int i = 0; i < book_count; i++) {
        if (strcmp(inventory[i].title, title) == 0) {
            for (int j = i; j < book_count - 1; j++) {
                inventory[j] = inventory[j + 1];
            }
            book_count--;
            printf("Book '%s' has been deleted from the library.\n", title);
            return;
        }
    }
    printf("Book '%s' not found in the library.\n", title);
}
```

• Borrow Book:

- I used (fgets) to read the book title from the user.
- I used a for loop to search through the inventory for the book title.
- I used the if statement with a condition that compares the title input by the user with the title of each book in the inventory
- If a match is found then the condition is true and another if statement will be checked to see if the book is borrowed or not.
- If the book is already borrowed a message will appear that the book is borrowed.
- If the book isn't borrowed then the book status will change in the inventory to be borrowed and a message will appear the says book is borrowed successfully.
- If a match isn't found then a message will appear that a book with this title isn't found in the library.

```
// Borrow book function:
void borrow_book() {
    char title[50];
    printf("\n--- Borrow Book ---\n");
    printf("Enter the title of the book to borrow: ");
    fgets(title, sizeof(title), stdin);
    title[strcspn(title, "\n")] = 0; // Remove newline character

    for (int i = 0; i < book_count; i++) {
        if (strcmp(inventory[i].title, title) == 0) {
            if (inventory[i].is_borrowed) {
                printf("Book '%s' is already borrowed.\n", title);
            } else {
                inventory[i].is_borrowed = 1;
                printf("Book '%s' has been borrowed successfully.\n", title);
            }
            return;
        }
    }
    printf("Book '%s' not found in the library.\n", title);
}
```

- Return Book:

- I used (fgets) to read the book title from the user.
- I used a for loop to search through the inventory for the book title.
- I used the if statement with a condition that compares the title input by the user with the title of each book in the inventory
- If a match is found then the condition is true and another if statement will be checked to see if the book is borrowed or not.
- If the book is marked as borrowed and can be returned , the book will be returned and the status of the book will be changed to not borrowed.
- If the book is marked as not borrowed and can't be returned , a message will appear that the book isn't borrowed.
- If a match isn't found then a message will appear that a book with this title isn't found in the library.

```
// Return book function:
void return_book() {
    char title[50];
    printf("\n--- Return Book ---\n");
    printf("Enter the title of the book to return: ");
    fgets(title, sizeof(title), stdin);
    title[strcspn(title, "\n")] = 0; // Remove newline character

    for (int i = 0; i < book_count; i++) {
        if (strcmp(inventory[i].title, title) == 0) {
            if (inventory[i].is_borrowed) {
                inventory[i].is_borrowed = 0;
                printf("Book '%s' has been returned successfully.\n", title);
            } else {
                printf("Book '%s' was not borrowed.\n", title);
            }
            return;
        }
    }
    printf("Book '%s' not found in the library.\n", title);
}
```

- Search Book:

```
// Search for a book function:
void search_book() {
    char title[50];
    printf("\n--- Search Book ---\n");
    printf("Enter the title of the book to search: ");
    fgets(title, sizeof(title), stdin);
    title[strcspn(title, "\n")] = 0; // Remove newline character

    for (int i = 0; i < book_count; i++) {
        if (strcmp(inventory[i].title, title) == 0) {
            printf("Book '%s' is found in the library.\n", title);
            return;
        }
    }
    printf("Book '%s' is not found in the library.\n", title);
}
```

- I used (fgets) to read the book title from the user.
- I used a for loop to search through the inventory for the book title.
- I used the if statement with a condition that compares the title input by the user with the title of each book in the inventory.
- If a match is found then the condition is true and a message will appear that the book is found in the library.
- If a match isn't found then a message will appear that a book with this title isn't found in the library.

- **Overdue Books:**

- I made three variables (day, month, year).
- I used (fgets) to read the book title from the user.
- I used the (scanf) to let the user input the day, the month and the year of borrowing.
- I used (time_t) to get the current system time, this is why I used the library (#include <time.h>).
- I made a struct to store in the borrowing date (tm borrowed_date).
- I setted the day of the borrowing date (tm_mday) to the day entered by the user, the month of borrowing date (tm_mon) to the month – 1 because (tm_mon) only expects months in the range 0–11 and the year of borrowing date (tm_year) to year – 1900 because the (tm_year) only expects years since 1900.
- Then converted the (borrowed_date) into seconds.
- Calculates the difference in seconds between the current time (t) and the borrowing time (borrowed_time), Converts the difference from seconds to days by dividing by the number of seconds in a day (60 * 60 * 24).
- I used an if condition to see if the difference is more than 14 days then a message will appear that this book is overdue and if the difference is less than 14 days then message will appear that this book isn't overdue.

```
// The overdue books function:
void overdue_books() {
    char title[50];
    int day, month, year;
    printf("\n--- Overdue Books ---\n");
    printf("Enter the title of the book: ");
    fgets(title, sizeof(title), stdin);
    title[strcspn(title, "\n")] = 0; // Remove newline character

    printf("Enter the date of borrowing (DD MM YYYY): ");
    scanf("%d %d %d", &day, &month, &year);

    // Get the current date
    time_t t = time(NULL);
    struct tm today = *localtime(&t);

    // Calculate the days since borrowing:
    struct tm borrowed_date = {0};
    borrowed_date.tm_mday = day;
    borrowed_date.tm_mon = month - 1; // Months are 0-11
    borrowed_date.tm_year = year - 1900; // Years since 1900
    time_t borrowed_time = mktime(&borrowed_date);

    double diff_in_days = difftime(t, borrowed_time) / (60 * 60 * 24);

    if (diff_in_days > 14) {
        printf("Book '%s' is overdue by %.0f days.\n", title, diff_in_days - 14);
    } else {
        printf("Book '%s' is not overdue.\n", title);
    }
}
```

- **Main Function:**

- I called the (initialize_inventory) function to put the book inventory in the system.
- I used a do, while loop to ensures that the menu will be displayed at least once and keeps looping until the user chooses to exit (choice 8).
- I called the (show_menu) to display the available options for the user.
- I used a switch case to let the user choose any option he wants from the functions provided in the menu.
- If the user input a number that isn't found in the cases then a message will appear saying "invalid choice .please try again".

```
// The Main function:
int main() {
    int choice;

    // Initialize inventory with some books
    initialize_inventory();

    do {
        show_menu();
        printf("Enter your choice: ");
        scanf("%d", &choice);
        getchar(); // Consume the newline left by scanf

        switch(choice) {
            case 1:
                view_inventory();
                break;
            case 2:
                add_book();
                break;
            case 3:
                delete_books();
                break;
            case 4:
                borrow_book();
                break;
            case 5:
                return_book();
                break;
            case 6:
                search_book();
                break;
            case 7:
                overdue_books();
                break;
            case 8:
                printf("Exiting the System. Goodbye!\n");
                break;
            default:
                printf("Invalid choice. Please try again.\n");
        }
    } while(choice != 8);

    return 0;
}
```

Screen shot of the program outputs:

1) The menu display:

```
Library Management System
1. View Books Inventory
2. Add Book
3. Delete Book
4. Borrow Book
5. Return Book
6. Search Book
7. View Overdue Books
8. Exit
Enter your choice: |
```

2) The Books Inventory:

```
Enter your choice: 1

--- Inventory ---
ID: 1, Title: Calculus 1, Author: Youssef Ahmed, Department: Math, Borrowed: No
ID: 2, Title: Computing, Author: Ahmed Mohamed, Department: Computer Science, Borrowed: No
ID: 3, Title: Calculus 2, Author: Ahmed Zaki, Department: Math, Borrowed: No
ID: 4, Title: Networks, Author: Mark Hany, Department: Computer Science, Borrowed: No
ID: 5, Title: Creativity, Author: Mona Ahmed, Department: Creative Thinking, Borrowed: No
ID: 6, Title: Information System, Author: Mahmoud Hassan, Department: Computer Science, Borrowed: No
ID: 7, Title: Database, Author: Rana Ahmed, Department: Computer Science, Borrowed: No
ID: 8, Title: Statics, Author: Mohamed Hassn, Department: Math, Borrowed: No
ID: 9, Title: Serway Physics, Author: Micheal Ben, Department: Physics, Borrowed: No
ID: 10, Title: Organic Chemistry, Author: Ben Pitt, Department: Chemistry, Borrowed: No
ID: 11, Title: Human Anatomy, Author: Allen Sorkin, Department: Biology, Borrowed: No
ID: 12, Title: Fundamentals Of Law, Author: Harvey Specter, Department: Law, Borrowed: No
ID: 13, Title: Dynamics, Author: Mike Ross, Department: Math, Borrowed: No
ID: 14, Title: Software Security, Author: Mohamed Seif, Department: Computer Science, Borrowed: No
ID: 15, Title: Plants Anatomy, Author: Rich Spencer, Department: Biology, Borrowed: No
```

3)Add book :

Inventory before adding:

```
Enter your choice: 2

--- Add Book ---
Enter the title of the book: Harry Potter
Enter the author of the book: Mark Sorkin
Enter the department of the book: Entertainment
Enter the ID of the book: 16
Book 'Harry Potter' by 'Mark Sorkin' added to the 'Entertainment' department with ID 16.
```

```
Enter your choice: 1

--- Inventory ---
ID: 1, Title: Calculus 1, Author: Youssef Ahmed, Department: Math, Borrowed: No
ID: 2, Title: Computing, Author: Ahmed Mohamed, Department: Computer Science, Borrowed: No
ID: 3, Title: Calculus 2, Author: Ahmed Zaki, Department: Math, Borrowed: No
ID: 4, Title: Networks, Author: Mark Hany, Department: Computer Science, Borrowed: No
ID: 5, Title: Creativity, Author: Mona Ahmed, Department: Creative Thinking, Borrowed: No
ID: 6, Title: Information System, Author: Mahmoud Hassan, Department: Computer Science, Borrowed: No
ID: 7, Title: Database, Author: Rana Ahmed, Department: Computer Science, Borrowed: No
ID: 8, Title: Statics, Author: Mohamed Hassn, Department: Math, Borrowed: No
ID: 9, Title: Serway Physics, Author: Micheal Ben, Department: Physics, Borrowed: No
ID: 10, Title: Organic Chemistry, Author: Ben Pitt, Department: Chemistry, Borrowed: No
ID: 11, Title: Human Anatomy, Author: Allen Sorkin, Department: Biology, Borrowed: No
ID: 12, Title: Fundamentals Of Law, Author: Harvey Specter, Department: Law, Borrowed: No
ID: 13, Title: Dynamics, Author: Mike Ross, Department: Math, Borrowed: No
ID: 14, Title: Software Security, Author: Mohamed Seif, Department: Computer Science, Borrowed: No
ID: 15, Title: Plants Anatomy, Author: Rich Spencer, Department: Biology, Borrowed: No
```

Inventory after adding:

```
--- Inventory ---
ID: 1, Title: Calculus 1, Author: Youssef Ahmed, Department: Math, Borrowed: No
ID: 2, Title: Computing, Author: Ahmed Mohamed, Department: Computer Science, Borrowed: No
ID: 3, Title: Calculus 2, Author: Ahmed Zaki, Department: Math, Borrowed: No
ID: 4, Title: Networks, Author: Mark Hany, Department: Computer Science, Borrowed: No
ID: 5, Title: Creativity, Author: Mona Ahmed, Department: Creative Thinking, Borrowed: No
ID: 6, Title: Information System, Author: Mahmoud Hassan, Department: Computer Science, Borrowed: No
ID: 7, Title: Database, Author: Rana Ahmed, Department: Computer Science, Borrowed: No
ID: 8, Title: Statics, Author: Mohamed Hassn, Department: Math, Borrowed: No
ID: 9, Title: Serway Physics, Author: Micheal Ben, Department: Physics, Borrowed: No
ID: 10, Title: Organic Chemistry, Author: Ben Pitt, Department: Chemistry, Borrowed: No
ID: 11, Title: Human Anatomy, Author: Allen Sorkin, Department: Biology, Borrowed: No
ID: 12, Title: Fundamentals Of Law, Author: Harvey Specter, Department: Law, Borrowed: No
ID: 13, Title: Dynamics, Author: Mike Ross, Department: Math, Borrowed: No
ID: 14, Title: Software Security, Author: Mohamed Seif, Department: Computer Science, Borrowed: No
ID: 15, Title: Plants Anatomy, Author: Rich Spencer, Department: Biology, Borrowed: No
ID: 16, Title: Harry Potter, Author: Mark Sorkin, Department: Entertainment, Borrowed: No
```

4)Delete book:

Inventory After deleting:

```
Enter your choice: 3

--- Delete Book ---
Enter the title of the book to delete: Harry Potter
Book 'Harry Potter' has been deleted from the library.
```

```
Enter your choice: 1

--- Inventory ---
ID: 1, Title: Calculus 1, Author: Youssef Ahmed, Department: Math, Borrowed: No
ID: 2, Title: Computing, Author: Ahmed Mohamed, Department: Computer Science, Borrowed: No
ID: 3, Title: Calculus 2, Author: Ahmed Zaki, Department: Math, Borrowed: No
ID: 4, Title: Networks, Author: Mark Hany, Department: Computer Science, Borrowed: No
ID: 5, Title: Creativity, Author: Mona Ahmed, Department: Creative Thinking, Borrowed: No
ID: 6, Title: Information System, Author: Mahmoud Hassan, Department: Computer Science, Borrowed: No
ID: 7, Title: Database, Author: Rana Ahmed, Department: Computer Science, Borrowed: No
ID: 8, Title: Statics, Author: Mohamed Hassn, Department: Math, Borrowed: No
ID: 9, Title: Serway Physics, Author: Micheal Ben, Department: Physics, Borrowed: No
ID: 10, Title: Organic Chemistry, Author: Ben Pitt, Department: Chemistry, Borrowed: No
ID: 11, Title: Human Anatomy, Author: Allen Sorkin, Department: Biology, Borrowed: No
ID: 12, Title: Fundamentals Of Law, Author: Harvey Specter, Department: Law, Borrowed: No
ID: 13, Title: Dynamics, Author: Mike Ross, Department: Math, Borrowed: No
ID: 14, Title: Software Security, Author: Mohamed Seif, Department: Computer Science, Borrowed: No
ID: 15, Title: Plants Anatomy, Author: Rich Spencer, Department: Biology, Borrowed: No
```

5) Borrow book:

```
Enter your choice: 4

--- Borrow Book ---
Enter the title of the book to borrow: Calculus 1
Book 'Calculus 1' has been borrowed successfully.
```

Inventory before and after borrowing:

```
ID: 1, Title: Calculus 1, Author: Youssef Ahmed, Department: Math, Borrowed: No
```

```
ID: 1, Title: Calculus 1, Author: Youssef Ahmed, Department: Math, Borrowed: Yes
```

6) Return book:

```
Enter your choice: 5

--- Return Book ---
Enter the title of the book to return: Computing
Book 'Computing' has been returned successfully.
```

Inventory before and after returning:

```
ID: 2, Title: Computing, Author: Ahmed Mohamed, Department: Computer Science, Borrowed: Yes
```

```
ID: 2, Title: Computing, Author: Ahmed Mohamed, Department: Computer Science, Borrowed: No
```

7) Search book:

Found:

```
Enter your choice: 6

--- Search Book ---
Enter the title of the book to search: Networks
Book 'Networks' is found in the library.
```

Not Found:

```
Enter your choice: 6

--- Search Book ---
Enter the title of the book to search: hdjdj
Book 'hdjdj' is not found in the library.
```

8) Overdue books:

Overdue:

```
Enter your choice: 7

--- Overdue Books ---
Enter the title of the book: Database
Enter the date of borrowing (DD MM YYYY): 1 12 2024
Book 'Database' is overdue by 6 days.
```

Not Overdue:

```
Enter your choice: 7

--- Overdue Books ---
Enter the title of the book: Database
Enter the date of borrowing (DD MM YYYY): 18 12 2024
Book 'Database ' is not overdue.
```


9)Exiting the system:

```
8. Exit
Enter your choice: 8
Exiting the System. Goodbye!

Process returned 0 (0x0)    execution time : 1960.102 s
Press any key to continue.
```