Name = Divya Kisan Gosavi
Std = MSC IMCA 1
Roll no.: 25092016
Subject = Data Structure

Library book management using Singly linked list in C

```
// Online C compiler to run C program online
#include <stdio.h>
#include <stdlib.h>

// Using typedef for cleaner syntax
typedef struct Book {
   int id;
   char name[50];
   char author[50];
   struct Book *next;
} Book;

// Head pointer (points to first book)
Book *head = NULL;

// Function to add a new book
void addBook() {
   Book *newBook = (Book*) malloc(sizeof(Book));
   printf("Enter Book ID: ");
   scanf("%d", &newBook->id);
   printf("Enter Book Name: ");
   scanf("%s", newBook->name);
   printf("Enter Author Name: ");
   scanf("%s", newBook->author);
   newBook->next = NULL;

   // If list is empty
   if (head == NULL) {
      head = newBook;
   } else {
      Book *temp = head;
      while (temp->next != NULL)
         temp = temp->next;
```

```
        temp->next = newBook;
    }
    printf("Book added successfully!\n");
}

// Function to display all books
void displayBooks() {
    Book *temp = head;
    if (temp == NULL) {
        printf("No books available!\n");
        return;
    }

    printf("\nLibrary Books:\n");
    while (temp != NULL) {
        printf("Book ID: %d\n", temp->id);
        printf("Book Name: %s\n", temp->name);
        printf("Author: %s\n\n", temp->author);
        temp = temp->next;
    }
}

// Function to search a book by ID
void searchBook() {
    int id;
    printf("Enter Book ID to search: ");
    scanf("%d", &id);

    Book *temp = head;
    while (temp != NULL) {
        if (temp->id == id) {
            printf("\nBook Found:\n");
            printf("Book ID: %d\n", temp->id);
            printf("Book Name: %s\n", temp->name);
            printf("Author: %s\n", temp->author);
            return;
        }
        temp = temp->next;
    }
    printf("Book not found!\n");
}
```

```
// Function to delete a book by ID
void deleteBook() {
    int id;
    printf("Enter Book ID to delete: ");
    scanf("%d", &id);

    Book *temp = head, *prev = NULL;

    // If head node itself holds the ID
    if (temp != NULL && temp->id == id) {
        head = temp->next;
        free(temp);
        printf("Book deleted successfully!\n");
        return;
    }

    // Search for the book
    while (temp != NULL && temp->id != id) {
        prev = temp;
        temp = temp->next;
    }

    // If not found
    if (temp == NULL) {
        printf("Book not found!\n");
        return;
    }

    // Unlink node and free memory
    prev->next = temp->next;
    free(temp);
    printf("Book deleted successfully!\n");
}

// Main Function
int main() {
    int choice;
    while (1) {
        printf("\n--- Library Book Management (SLL using typedef) ---\n");
        printf("1. Add Book\n");
```

```
        printf("2. Display Books\n");
        printf("3. Search Book\n");
        printf("4. Delete Book\n");
        printf("5. Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
            case 1: addBook(); break;
            case 2: displayBooks(); break;
            case 3: searchBook(); break;
            case 4: deleteBook(); break;
            case 5: exit(0);
            default: printf("Invalid choice!\n");
        }
    }
    return 0;
}
```