

S.No: 1	Exp. Name: <i>Project Module</i>	Date: 2024-06-14
---------	---	------------------

Aim:

Project Module

Source Code:

```
hello.c
```

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

typedef struct User {
    char username[50], password[50];
    struct User* next;
} User;

typedef struct Movie {
    int id, availableSeats;
    char name[100], genre[50];
    struct Movie* next;
} Movie;

typedef struct Booking {
    int bookingId, movieId, seatsBooked;
    char username[50];
    struct Booking* next;
} Booking;

User *userHead = NULL;
Movie *movieHead = NULL;
Booking *bookingHead = NULL;
int bookingCount = 0;

void flushInput() {
    int c;
    while ((c = getchar()) != '\n' && c != EOF);
}

void registerUser() {
    User* newUser = (User*)malloc(sizeof(User));
    printf("Enter username: ");
    scanf("%49s", newUser->username);
    for (User* temp = userHead; temp; temp = temp->next) {
        if (!strcmp(temp->username, newUser->username)) {
            printf("Username exists!\n");
            free(newUser);
            return;
        }
    }
    printf("Enter password: ");
    scanf("%49s", newUser->password);
    newUser->next = userHead;
    userHead = newUser;
}

```

```

        printf("User registered successfully!\n");
    }

void loginUser() {
    char username[50], password[50];
    printf("Enter username: ");
    scanf("%49s", username);
    printf("Enter password: ");
    scanf("%49s", password);
    for (User* temp = userHead; temp; temp = temp->next) {
        if (!strcmp(temp->username, username) && !strcmp(temp->password, password)) {
            printf("Login successful!\n");
            return;
        }
    }
    printf("Invalid username or password!\n");
}

void addMovie() {
    Movie* newMovie = (Movie*)malloc(sizeof(Movie));
    newMovie->id = rand();
    printf("Enter movie name: ");
    flushInput();
    fgets(newMovie->name, sizeof(newMovie->name), stdin);
    newMovie->name[strcspn(newMovie->name, "\n")] = '\0';
    printf("Enter genre: ");
    scanf("%49s", newMovie->genre);
    printf("Enter available seats: ");
    scanf("%d", &newMovie->availableSeats);
    newMovie->next = movieHead;
    movieHead = newMovie;
    printf("Movie added successfully!\n");
}

void viewMovies() {
    if (!movieHead) {
        printf("No movies available.\n");
        return;
    }
    for (Movie* temp = movieHead; temp; temp = temp->next) {
        printf("Movie ID: %d\nName: %s\nGenre: %s\nAvailable Seats: %d\n\n", temp->id, temp->name, temp->genre, temp->availableSeats);
    }
}

```

```

void bookTicket() {
    char username[50];
    int movieId, seatsBooked;
    printf("Enter username: ");
    scanf("%49s", username);
    printf("Enter movie ID: ");
    scanf("%d", &movieId);
    printf("Enter number of seats: ");
    scanf("%d", &seatsBooked);
    for (Movie* movie = movieHead; movie; movie = movie->next) {
        if (movie->id == movieId && movie->availableSeats >=
seatsBooked) {
            Booking* newBooking =
(Booking*)malloc(sizeof(Booking));
            newBooking->bookingId = ++bookingCount;
            strcpy(newBooking->username, username);
            newBooking->movieId = movieId;
            newBooking->seatsBooked = seatsBooked;
            newBooking->next = bookingHead;
            bookingHead = newBooking;
            movie->availableSeats -= seatsBooked;
            printf("Booking successful! Your booking ID is %d\n",
bookingCount);
            return;
        }
    }
    printf("Booking failed! Not enough available seats.\n");
}

void validateEntry() {
    int bookingId;
    printf("Enter booking ID: ");
    scanf("%d", &bookingId);
    for (Booking* booking = bookingHead; booking; booking =
booking->next) {
        if (booking->bookingId == bookingId) {
            printf("Entry validated successfully!\n");
            return;
        }
    }
    printf("Invalid booking ID!\n");
}

void viewBookings() {
    if (!bookingHead) {

```

```

        printf("No bookings available.\n");
        return;
    }
    for (Booking* temp = bookingHead; temp; temp = temp->next) {
        printf("Booking ID: %d\nUsername: %s\nMovie ID: %d\nSeats
Booked: %d\n\n", temp->bookingId, temp->username, temp->movieId,
temp->seatsBooked);
    }
}

int main() {
    // Register user outside the loop
    printf("User Registration:\n");
    registerUser();

    int choice;
    while (1) {
        printf("\nMovie Ticket Sales and Theater Entry Management
System\n");
        printf("1. Login\n2. Add Movie (Admin)\n3. View Movies\n4.
Book Ticket\n5. Validate Entry\n6. View Bookings\n7. Exit\nEnter
your choice: ");
        if (scanf("%d", &choice) != 1) {
            printf("Invalid input! Please enter a number.\n");
            flushInput();
            continue;
        }
        switch (choice) {
            case 1:
                loginUser();
                break;
            case 2:
                addMovie();
                break;
            case 3:
                viewMovies();
                break;
            case 4:
                bookTicket();
                break;
            case 5:
                validateEntry();
                break;
            case 6:
                viewBookings();
                break;
        }
    }
}

```

```
        case 7:
            exit(0);
        default:
            printf("Invalid choice!\n");
    }
}
```

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Hello World