

The background features a light gray illustration of a vintage movie camera on a tripod, centered behind the text. Two red film strips are also visible: one in the top-left corner curving downwards, and another in the bottom-right corner curving upwards.

Movie Ticket

Booking System

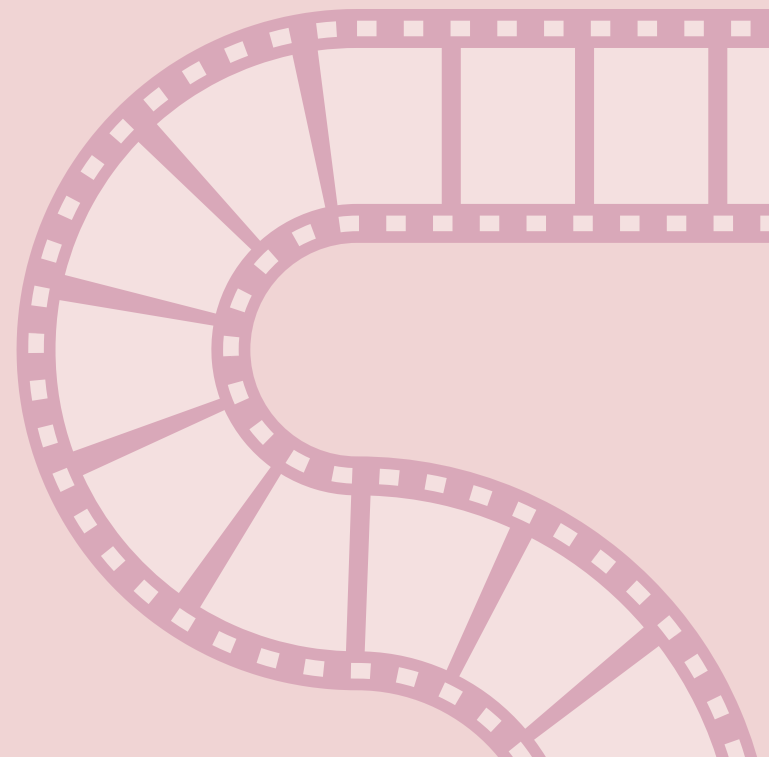


Agenda

To create an interactive movie ticket booking system using C++ with features like seat selection, pricing, and booking management.



Key Features

- 1 Seamless Experience
 - 2 Accurate Seat Location
 - 3 Dynamic Pricing
- 

System Workflow



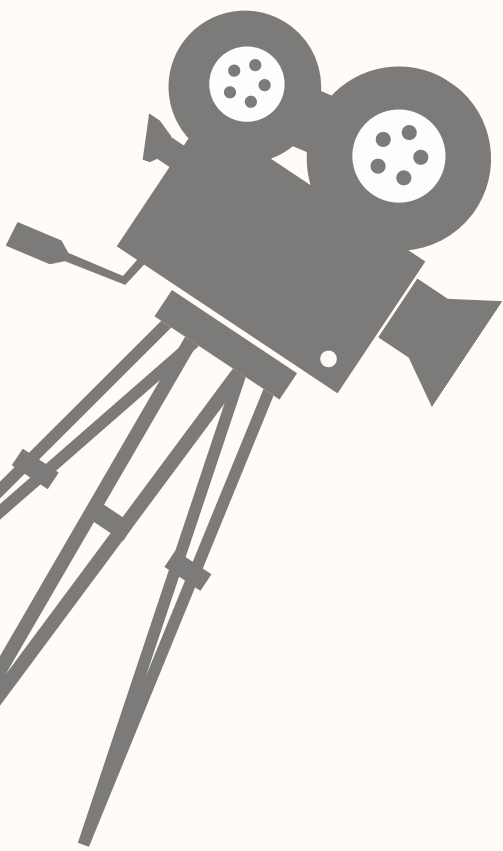
Displays Available Movies

Booking Preference

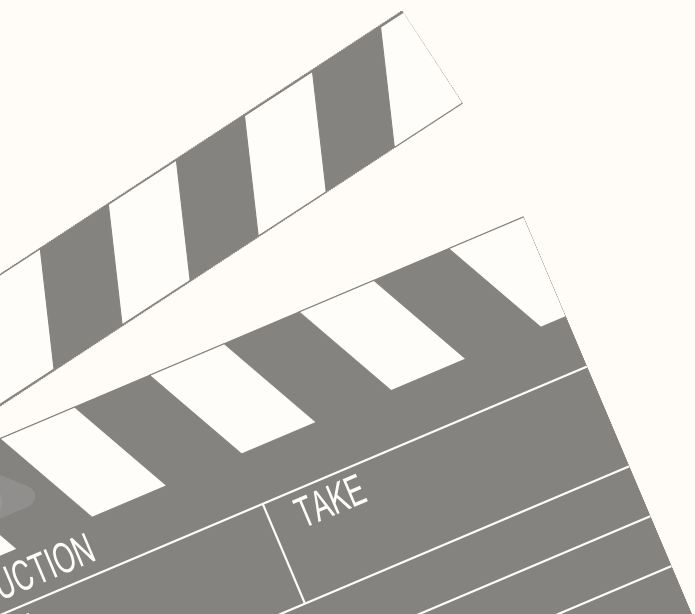
Class/Seat Selection

Booking Summary

Exploring Further



Core Functionality



Checks for movie
availability

Provides different
Theater options

Seat reservation with
conflict handling

Option to view
upcoming movies.



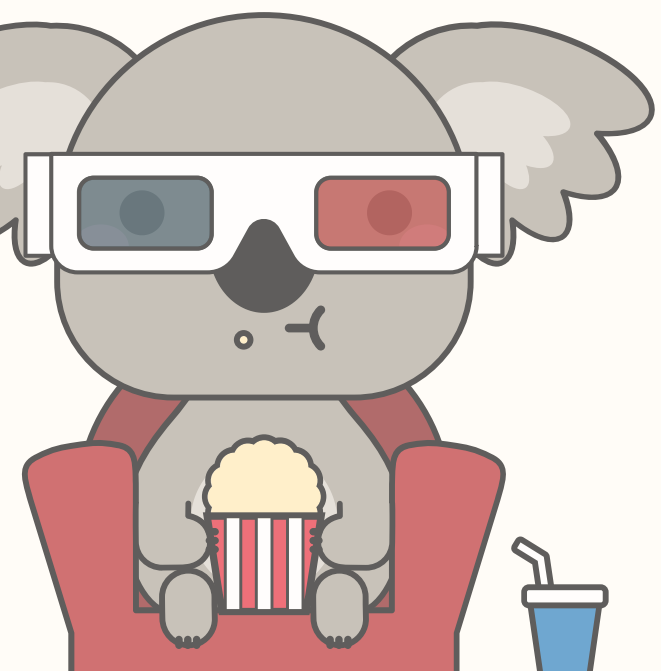
Classes and Functions

Classes

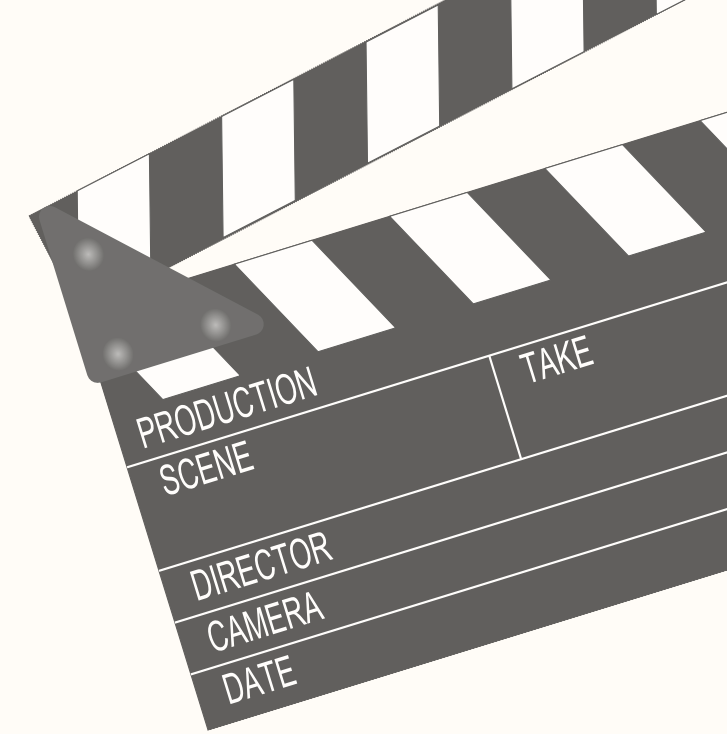
- 1 Movie: Stores movie details and availability.
- 2 Ticket (Abstract): Base class for seat classes.
- 3 PlatinumTicket and GoldTicket: Derived classes for pricing and seat limits.

Utility Functions

- 1 toLowerCase(): Case-insensitive string comparison.
- 2 displayOptions(): Lists options interactively.

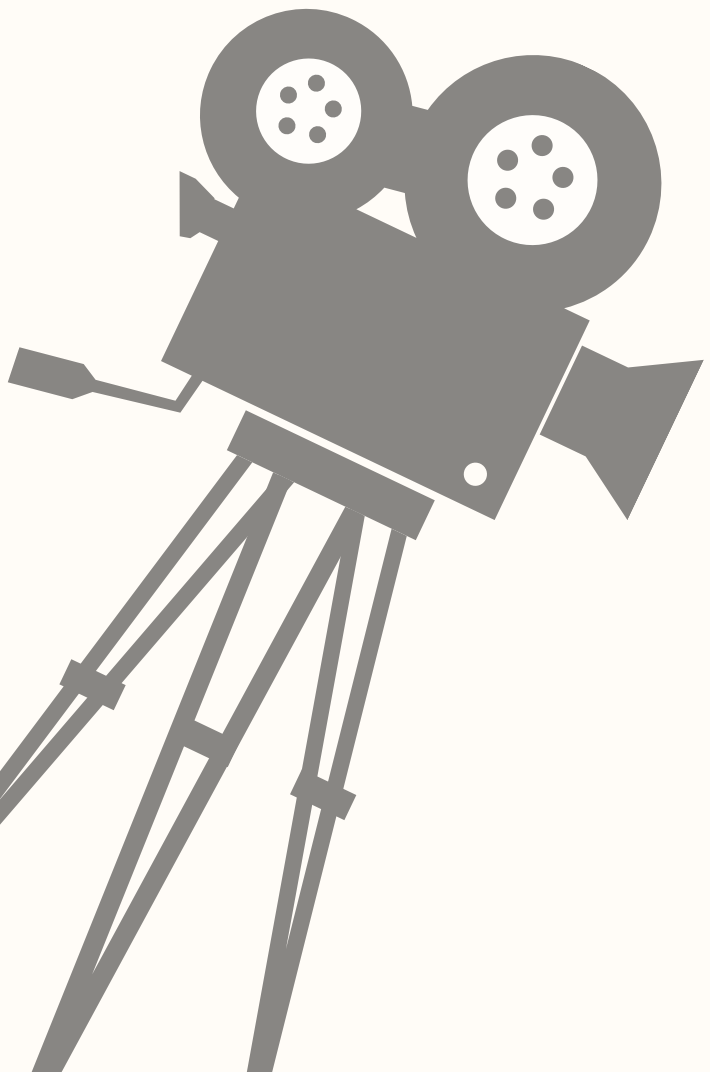


Code-snippets



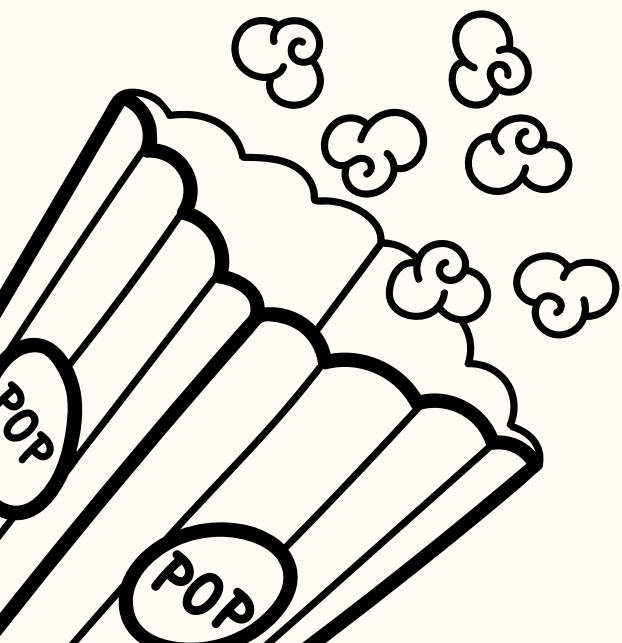
Movie Availability:

```
class Movie {  
public:  
    string name;  
    Movie(const string& name) : name(name) {}  
    static bool isMovieAvailable(const vector<Movie>& movies, const string& movieName) {  
        string lowerMovieName = toLowerCase(movieName);  
        for (const auto& movie : movies) {  
            if (toLowerCase(movie.name) == lowerMovieName) {  
                return true;  
            }  
        }  
        return false;  
    }  
};
```



Code-snippets

Seat Selection:



```
int startSeat, endSeat;
while (true) {
    cout << "Enter the seat number or range (e.g., 4 for a single seat or 4 10 for a range): ";
    cin >> startSeat;
    if (cin.peek() == '\n') {
        endSeat = startSeat;
    } else {
        cin >> endSeat;
    }

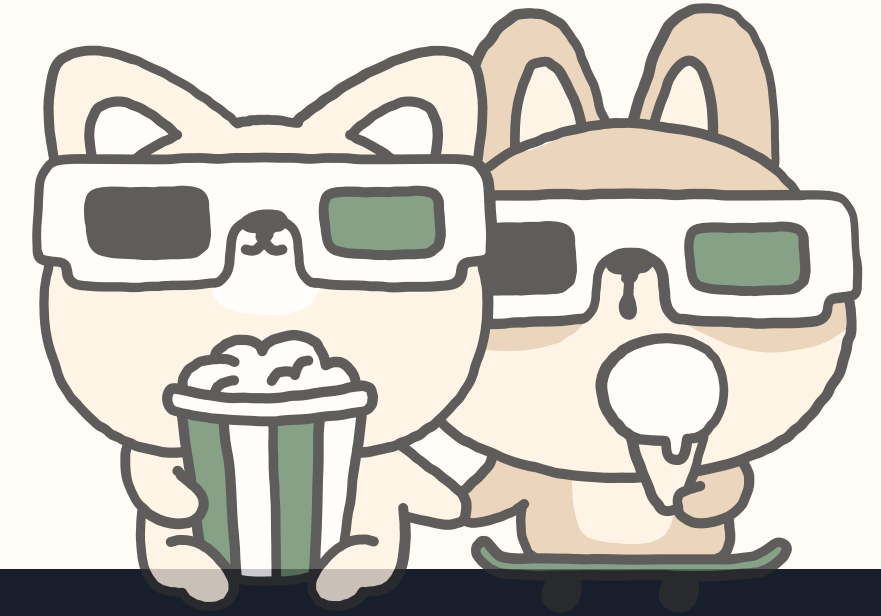
    if (startSeat < 1 || endSeat > ticket->getMaxSeats() || startSeat > endSeat) {
        cout << "Invalid seat range! Please enter a valid range within 1 to " << ticket
            ->getMaxSeats() << "." << endl;
        continue;
    }

    Booking* existingBooking = findBooking(dateStr, selectedTheater, selectedTiming, movieName
        );

    if (!existingBooking) {
        bookings.push_back({dateStr, selectedTheater, selectedTiming, movieName, {}});
        existingBooking = &bookings.back();
    }
}
```

Code-snippets

Seat Selection:



```
bool seatsAvailable = true;
for (int seat = startSeat; seat <= endSeat; ++seat) {
    string seatId = selectedRow + to_string(seat);
    if (existingBooking->bookedSeats.count(seatId)) {
        seatsAvailable = false;
        break;
    }
}

if (!seatsAvailable) {
    cout << "Some seats in the range are already booked for this date, theater, and timing.
    Please try another range." << endl;
    continue;
}

for (int seat = startSeat; seat <= endSeat; ++seat) {
    existingBooking->bookedSeats.insert(selectedRow + to_string(seat));
}
break;
}
```



Booking Information



```
Booking Summary:  
Movie: ka  
Date: 18 November  
Theater: Mythri Cinemas  
Timing: 10:00 AM  
Class: gold  
Row: B  
Seats: 1 to 5  
Total Cost: 975 rupees
```

At the end the code will display the entire booking summary

Thank you!

Done By:

Shriya Rao Balivada(AP23110010540)

Venkata Asha Bathula (AP23110010550)

Kalisetty Manjari(AP23110010567)

Aravelli Jahnavi Jyotirmayi(AP23110010506)