

# **MOVIE TICKET BOOKING SYSTEM**



## **A PROJECT REPORT**

*Submitted by*

**SUGAPRIYA A (2303811724322112)**

*In partial fulfillment of requirements for the award of the course*

**CGB1201 – JAVAPROGRAMMING**

*in*

**ARTIFICIAL INTELLIGENCE AND DATA SCIENCE**

**K.RAMAKRISHNAN COLLEGE OF TECHNOLOGY**

(An Autonomous Institution , affiliated to Anna University Chennai and Approved  
by AICTE, New Delhi)

**SAMAYAPURAM-621112**

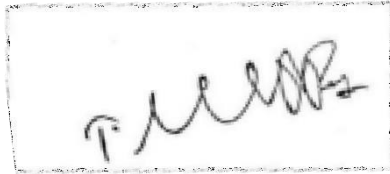
**DECEMBER,2024**

**K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY (AUTONOMOUS)**

**SAMAYAPURAM-621112**

**BONAFIDE CERTIFICATE**

Certified that this project report on “**MOVIE TICKET BOOKING SYSTEM**” is the bonafide work of **SUGAPRIYA A (2303811724322112)** who carried out the project work during the academic year 2024-2025 under my supervision.



Signature

Dr.T.AVUDAIAPPAN M.E.,Ph.D.,

**SUPERVISOR,**

Department of Artificial Intelligence,  
K. Ramakrishnan College of Technology,  
Samayapuram,Trichy-621112.



Signature

Mrs.S.GEETHA M.E.,

**HEAD OF THE DEPARTMENT,**

Department of Artificial Intelligence,  
K. Ramakrishnan College of Technology,  
Samayapuram,Trichy-621112.

Submitted for the viva-voce examination held on 3.12.24



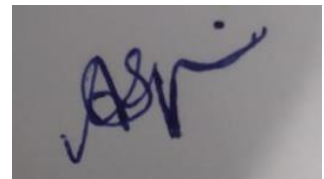
**INTERNAL EXAMINER**



**EXTERNAL EXAMINER**

## DECLARATION

I declare that the project report on “**MOVIE TICKET BOOKING SYSTEM**” is the result of original work done by me and best of our knowledge, similar work has not been submitted to “**ANNA UNIVERSITY CHENNAI**” for the requirement of Degree of **BACHELOR OF TECHNOLOGY**. This project report is submitted on the partial fulfillment of the requirement of the award of the **CGB1201–JAVAPROGRAMMING**.

A handwritten signature in blue ink, appearing to read 'ASV' with a flourish, on a light-colored background.

**Signature**

**SUGAPRIYA A**

**Place:** Samayapuram

**Date:** 3/12/2024

## ACKNOWLEDGEMENT

It is with great pride that I express our gratitude and indebtedness to our institution, **“K.Ramakrishnan College of Technology (Autonomous)”**, for providing us with the opportunity to do this project.

I extend our sincere acknowledgment and appreciation to the esteemed and honourable Chairman, **Dr.K.RAMAKRISHNAN, B.E.**, for having provided the facilities during the course of our study in college.

I would like to express our sincere thanks to our beloved Executive Director, **Dr. S.KUPPUSAMY, MBA, Ph.D.**, for forwarding our project and offering an adequate duration to complete it.

I would like to thank **Dr. N. VASUDEVAN, M.TECH., Ph.D.**, Principal, who gave the opportunity to frame the project to full satisfaction.

I thank **Dr.T.AVUDAIAPPAN, M.E., Ph.D.**, Head of the Department of **ARTIFICIAL INTELLIGENCE AND DATA SCIENCE**, for providing his encouragement in pursuing this project.

I wish to convey our profound and heartfelt gratitude to our esteemed project guide **Mrs.S.GEETHA M.E.**, Department of **ARTIFICIAL INTELLIGENCE AND DATA SCIENCE**, for her incalculable suggestions, creativity, assistance and patience, which motivated us to carry out this project.

I render our sincere thanks to the Course Coordinator and other staff members for providing valuable information during the course.

I wish to express our special thanks to the officials and Lab Technician so four departments who rendered their help during the period of the work progress.

## **VISION OF THE INSTITUTION**

To serve the society by offering top-notch technical education on par with Global standards.

## **MISSION OF THE INSTITUTION**

- Be a centre of excellence for technical education in emerging technologies by exceeding the needs of industry and society.
- Be an institute with world class research facilities.
- Be an institute nurturing talent and enhancing competency of students to transform them as all-round personalities respecting moral and ethical values.

## **VISION AND MISSION OF THE DEPARTMENT**

To excel in education, innovation and research in Artificial Intelligence and Data Science to fulfill industrial demands and societal expectations.

Mission1: To educate future engineers with solid fundamentals, continually improving teaching methods using modern tools.

Mission2: To collaborate with industry and offer top-notch facilities in a conducive learning environment.

Mission3: To foster skilled engineers and ethical innovation in AI and Data Science for global recognition and impactful research.

Mission4: To tackle the societal challenge of producing capable professionals by in stilling employ ability skills and human values.

## **PROGRAM EDUCATIONAL OBJECTIVES (PEOS)**

**PEO1:** Compete on a global scale for a professional career in Artificial Intelligence and Data Science.

**PEO2:** Provide industry-specific solutions for the society with effective communication and ethics.

**PEO3:** Hone their professional skills through research and lifelong learning initiatives.

## **PROGRAM OUTCOMES**

Engineering students will be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multi disciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multi disciplinary environments.
12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

#### **PROGRAM SPECIFIC OUTCOMES (PSOs)**

- **PSO 1:** Capable of working on data-related methodologies and providing industry-focused solutions.
- **PSO2:** Capable of analyzing and providing a solution to a given real-world problem by designing an effective program.

## **ABSTRACT**

This project presents a Movie Ticket Booking System implemented in Java using the Abstract Window Toolkit (AWT). The application provides an interactive graphical user interface (GUI) for managing movie schedules, seat availability, ticket bookings, and reservation retrievals. It employs object-oriented principles for modularity, with dedicated classes to handle movies and reservations. Efficient data management is achieved through collections like HashMap, while event-driven programming ensures responsiveness and interactivity. The program validates user input, updates seat availability dynamically, and generates unique reservation IDs. Designed with scalability in mind, the system allows for potential future enhancements such as database integration and additional user functionalities. This project serves as a practical example of applying core Java concepts to solve real-world problems, making it an excellent tool for learning and application development.



## TABLE OFCONTENTS

CHAPTER No.	TITLE	PAGE No.
	<b>ABSTRACT</b>	<b>viii</b>
<b>1</b>	<b>INTRODUCTION</b>	<b>1</b>
	1.1 INTRODUCTION	1
	1.2 OBJECTIVE	1
<b>2</b>	<b>PROJECT METHODOLOGY</b>	<b>2</b>
	2.1 PROPOSEDWORK	2
	2.2 BLOCKDIAGRAM	2
<b>3</b>	<b>JAVA PROGRAMMING CONCEPTS</b>	<b>3</b>
	3.1 AWT(ABSTRACTWINDOWTOOLKIT)	3
	3.2 EVENT HANDLING	3
	3.3 OOP CONCEPT	3
	3.4 COLLECTION FRAMEWORK	3
<b>4</b>	<b>MODULE DESCRIPTION</b>	<b>4</b>
	4.1GUI SETUP MODULE	4
	4.2 MOVIE MANAGEMENT MODULE	4
	4.3TICKET BOOKING MODULE	4
	4.4 RESERVATION MANAGEMENT MODULE	4
	4.5 MOVIEDISPLAYMODULE	4
<b>5</b>	<b>CONCLUSION</b>	<b>5</b>
	<b>REFERENCES</b>	<b>6</b>
	<b>APPENDICES</b>	<b>7</b>
	AppendixA–Sourcecode	7
	AppendixB–Screenshots	17

# **CHAPTER 1**

## **INTRODUCTION**

### **1.1 INTRODUCTION**

The Movie Ticket Booking System is a Java-based application designed to provide a simple and efficient platform for managing movie schedules, seat bookings, and reservation tracking. Built using the Abstract Window Toolkit(AWT),the application offers an interactive graphical user interface (GUI) that enables users to select movies, choose show times, book tickets, and view reservation details. By integrating essential programming concepts such as object-oriented principles, event-driven programming, and data management with collections, the system demonstrate show Java can be utilized to create user-friendly and functional applications. This project aims to simulate a real-world ticket booking system while serving as a learning tool for Java developers.

### **1.2 OBJECTIVE**

The primary objective of this project is to develop a user-friendly and efficient Movie Ticket Booking System using Java. The system aims to stream line the process of selecting movies, choosing show times, booking tickets, and managing reservations through an interactive graphical user interface (GUI). By implementing core Java concepts such as object-oriented programming, event-driven programming, and efficient data handling, the project seeks to provide a practical application that simulates real-world ticket booking scenarios. Additionally, it serves as an educational tool for understanding Java application development and lays the groundwork for further enhancements, such as database integration and advanced user management features.

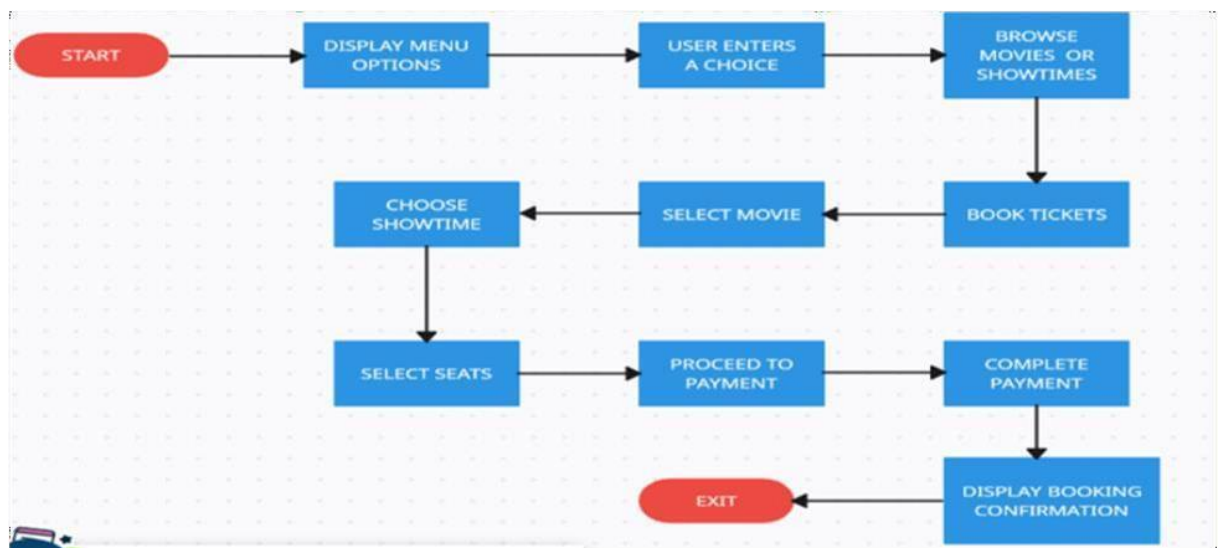
## CHAPTER 2

### PROJECTMETHODOLOGY

#### 2.1 PROPOSED WORK

The proposed Movie Ticket Booking System aims to create an interactive and efficient platform for managing movie schedules, ticket bookings, and reservations using Java's AWT for the graphical user interface. The system will allow users to select movies, choose show times, book tickets, and view reservations with real-time updates on seat availability. Key features include movie management, dynamic show time updates, unique reservation ID generation, and a display of available movies and seats. The application will incorporate input validation and feedback mechanisms to enhance user experience and ensure error-free functionality. Designed with scalability in mind, the system provides a foundation for future enhancements, such as integrating payment systems or database storage.

#### 2.2 BLOCKDIAGRAM



## **CHAPTER 3**

### **JAVA PROGRAMMING CONCEPTS**

#### **3.1 AWT (ABSTRACT WINDOW TOOLKIT):**

- GUI components like Frame, Button, Choice, TextField, TextArea, Label, and layout management using Flow Layout.

#### **3.2 EVENT HANDLING:**

- Action Listen errand Item Listener for handling button clicks and drop down selection changes.
- Window Adapter for handling window closing events.

#### **3.3 OOP CONCEPTS:**

- **Classes and Objects:** Movie and Reservation classes are used to encapsulate movie and reservation details.
- **Encapsulation:** Fields in Movie and Reservation are kept private (with in the class), and constructors are used to initialize them.
- **Inheritance:** The Movie Ticket Booking AWT class extends the Frame class to create a custom GUI frame.

#### **3.4 COLLECTIONS FRAMEWORK:**

- HashMap is used to store movies and reservations, enabling quick look up and insertion operations.

## **CHAPTER 4**

### **MODULE DESCRIPTION**

#### **4.1 GUI SETUP MODULE**

- Sets up the AWT components like buttons, text fields, choice lists, and labels.
- Adds components to the frame and sets the layout using Flow Layout.

#### **4.2 MOVIE MANAGEMENT MODULE**

- `initializeMovies()`: Initializes the movie data (names, showtimes, and available seats).
- `updateShowTimes()`: Updates the showtimes drop down when a movie is selected.

#### **4.3 TICKET BOOKING MODULE**

- `bookTickets()`: Validates input, checks seat availability, books tickets, generates a unique reservation ID, and updates the movie data.

#### **4.4 RESERVATION MANAGEMENT MODULE**

- `viewReservation()`: Retrieves and displays reservation details using the reservation ID.

#### **4.5 MOVIE DISPLAY MODULE**

- `showMovies()`: Displays all available movies, their show times, and the number of available seats.

## **CHAPTER 5**

### **CONCLUSION**

The Movie Ticket Booking System effectively demonstrates the application of core Java programming concepts to solve practical problems with a user-centric approach. By utilizing AWT for GUI development, the program provides an intuitive interface for tasks such as selecting movies, booking tickets, and managing reservations. The integration of object-oriented principles, event-driven programming, and efficient data handling through collections ensures a robust and maintainable design. The system's features, such as real-time seat availability updates, reservation retrieval, and dynamic showtime selection, make it both functional and engaging for users.

This project serves as a foundation for learning Java application development and can be extended to meet more complex requirements. Future enhancements, such as database integration, online payment systems, and multi-user access, can be incorporated to make the system suitable for larger-scale, real-world scenarios. Overall, the program demonstrates how Java can be effectively used to create scalable, modular, and interactive applications, providing valuable insights for both beginners and experienced developers.

## **REFERENCES:**

### **Java Tutorials:**

Comprehensive tutorials on Java programming, covering GUI development, object-oriented principles, and event-driven programming.

Website : <https://www.javatpoint.com/java-tutorial>

### **Java Swing and AWT Guides:**

Tutorials on creating graphical user interfaces in Java using AWT and Swing.

Website : <https://www.geeksforgeeks.org/awt-in-java/>

### **Java Documentation:**

The official Oracle Java documentation provides detailed information on Java programming concepts, including AWT, event handling, and collections.

Website : <https://docs.oracle.com/javase/>

## APPENDICES A - SOURCECODE

```
import
java.awt.*;import java.a
wt.event.*;
import java.util.HashMap;
import java.util.Map;
public class MovieBooking extends JFrame {
    static final int TICKET_PRICE = 200; // Ticket price per seat

    // Data storage for movies, reservations, and payment statuses
    static Map<String, Movie> movies = new HashMap<>();
    static Map<String, Reservation> reservations = new HashMap<>();

    // GUI Components
    Choice movieChoice, timeChoice, paymentMethodChoice;
    TextField seatsField, reservationIdField, nameField, paymentDetailsField;
    TextArea statusArea;
    Button bookButton, viewReservationButton, showMoviesButton, makePaymentBu
tton;

    // Movie class to store
    movie details
    static class Movie {
        String
        name;
        String[] show
        Times;
        int[] available
        Seats;

        Movie(String name, String[] showTimes, int[] availableSeats) {
```



```

        this.name
        =name;this.showTimes=show
        Times;
        this.availableSeats=availableSeats;
    }
}

//Reservationclasstostorerreservationdetailsstatic
classReservation{
    StringreservationId;
    String
    movieName;String
    showTime;intseatsB
    ooked;
    int
    amountToPay;Stringc
    ustomerName;boolea
    nisPaid;
    Reservation(StringreservationId,StringmovieName,StringshowTime,intseatsBooked,
StringcustomerName) {
        this.reservationId=reservationId;t
        his.movieName =
        movieName;this.showTime=sho
        wTime;this.seatsBooked=seatsBo
        oked;
        this.amountToPay=seatsBooked *TICKET_PRICE;//Calculatetotal
amount
        this.customerName=customerName;t
        his.isPaid=false;//Default:unpaid
    }
}

```

```

//ConstructortoinitalizeGUIpublic
MovieBooking(){
    setTitle("MovieTicketBookingSystem");s
    etSize(600,600);
    setLayout(newFlowLayout());

    //MovieSelection
    add(newLabel("SelectMovie:"));
    movieChoice = new
    Choice();add(movieChoice);

    //ShowTimeSelection
    add(newLabel("SelectShowTime:"));ti
    meChoice =
    newChoice();add(timeChoice);

    //SeatsField
    add(newLabel("EnterNumberofSeats:"));se
    atsField=new
    TextField(10);add(seatsField);

    //CustomerName Field
    add(newLabel("EnterYourName:"));n
    ameField = new
    TextField(20);add(nameField);

    // Buttons for booking and showing
    moviesbookButton=newButton("BookTick
    ets");
    viewReservationButton=newButton("ViewReservation");

```

```
showMoviesButton=newButton("ShowAvailableMovies");add(bookButton);  
add(viewReservationButton);  
add(showMoviesButton);
```

```
//ReservationID Field  
add(newLabel("EnterReservationID:"));r  
reservationIdField = new  
TextField(15);add(reservationIdField);
```

```
//PaymentSection  
add(newLabel("PaymentMethod:"));paymen  
tMethodChoice = new  
Choice();paymentMethodChoice.add("Credi  
t Card");paymentMethodChoice.add("Debit  
Card");paymentMethodChoice.add("UPI");  
paymentMethodChoice.add("Net  
Banking");add(paymentMethodChoice);
```

```
add(newLabel("EnterPaymentDetails:"));p  
aymentDetailsField = new  
TextField(20);add(paymentDetailsField);
```

```
makePaymentButton=newButton("MakePayment");add  
(makePaymentButton);
```

```
//StatusArea  
statusArea=newTextArea(10,50);st  
atusArea.setEditable(false);
```

```

add(statusArea);

//AddListeners
bookButton.addActionListener(e-
>bookTickets());viewReservationButton.addActionListener(e-
>viewReservation());showMoviesButton.addActionListener(e-
>showMovies());makePaymentButton.addActionListener(e->processPayment());

addWindowListener(new WindowAdapter()
    {publicvoidwindowClosing(WindowEvent){
        System.exit(0);
    }
});

//Initializemoviesi
nititalizeMovies();
}

//Methodtoinitialize
moviesandtheirshowtimesvoidinitializeMovies(){
    movies.put("Black",newMovie("Black",newString[]{"10:00AM","1:00PM","
4:00 PM"},newint[]{50,50,50}));
    movies.put("JurassicPark",newMovie("JurassicPark",newString[]{"11:00AM"
,"2:00 PM","5:00 PM"},newint[]{50,50,50}));
    movies.put("Veronica",newMovie("Veronica",newString[]{"12:00PM","3:00
PM","6:00 PM"},newint[]{50,50,50}));

    for(StringmovieName:movies.keySet()){movieChoice.a
        dd(movieName);

```

```

    }

    movieChoice.addItemListener(e->updateShowTimes());updateShowTimes();//Initializeshowtimesforthefirstmovie
}

//UpdateshowtimesinthetimeChoicedropdownbasedonselectedmovievoidupdateShowTimes() {
    timeChoice.removeAll();
    StringselectedMovie=movieChoice.getSelectedItem();Movie movie =movies.get(selectedMovie);
    for(Stringtime:movie.showTimes){timeChoice.add(time);
    }
}

//MethodtobookticketsvoidbookTickets(){
    StringselectedMovie=movieChoice.getSelectedItem();StringselectedTime =
    timeChoice.getSelectedItem();StringcustomerName =
    nameField.getText();
    intseatsRequested;

    try{
        seatsRequested=Integer.parseInt(seatsField.getText());
    }catch(NumberFormatException){
        statusArea.setText("Invalidnumberofseats.Pleaseenteravalidnumber.");
        return;
    }
}

```

```

    }

    Movie movie = movies.get(selectedMovie);
    int showIndex = timeChoice.getSelectedIndex();

    if (movie.availableSeats[showIndex] < seatsRequested)
    {
        statusArea.setText("Not enough seats available for this show.");
        return;
    }

    //Generate a unique reservation ID
    String reservationId = "RES" + (reservations.size() + 1);
    Reservation reservation = new Reservation(reservationId, selectedMovie, selectedTime, seatsRequested, customerName);
    reservations.put(reservationId, reservation);

    //Deduct seats
    movie.availableSeats[showIndex] -= seatsRequested;

    statusArea.setText("Bookings successful!\nReservation ID: " + reservationId
        + "\n" +
        "Movie: " + selectedMovie + "\nShow Time: " + selectedTime + "\nSeats
        Booked: " + seatsRequested +
        "\nAmount to Pay: ₹" + reservation.amountToPay + "\nPayment
        Pending");
    }

    //Method to process payment via
    idprocessPayment() {

```

```

StringreservationId=reservationIdField.getText();

if (!reservations.containsKey(reservationId))
    {statusArea.setText("Reservationnotfound!Pleaseenteravalid
ReservationID.");
    return;
}

Reservationreservation=reservations.get(reservationId);if(
reservation.isPaid){
    statusArea.setText("Paymentalreadycompletedforthisreservation.");return;
}

StringpaymentMethod=paymentMethodChoice.getSelectedItem();Stringp
aymentDetails=paymentDetailsField.getText();

if(paymentDetails.isEmpty()){
    statusArea.setText("Pleaseenterpaymentdetailstoproceed.");retu
    rn;
}

//Simulatepaymentprocessingr
eservation.isPaid=true;
statusArea.setText("PaymentSuccessful!\nReservationID:"+reservationId
+"\n"+
        "AmountPaid:₹"+reservation.amountToPay+"\nPaymentMethod:
"+paymentMethod);

```

```
}
```

```
//Method to view reservation details void
```

```
idviewReservation(){
```

```
    String reservationId=reservationIdField.getText();
```

```
    if (!reservations.containsKey(reservationId))
```

```
        { statusArea.setText("Reservation not found!");r
```

```
        eturn;
```

```
}
```

```
Reservation reservation=reservations.get(reservationId);statusArea.setText("Reservation Details:\n"+
```

```
    "ReservationID:"+reservation.reservationId+"\n"+"Movie:"+r
```

```
    reservation.movieName +"\n"+
```

```
    "Show Time: " + reservation.showTime + "\n"
```

```
    +"Seats Booked:"+reservation.seatsBooked+"\n"+
```

```
    "Amount to Pay: ₹" + reservation.amountToPay + "\n"
```

```
    +"Customer Name: " + reservation.customerName + "\n"
```

```
    +"PaymentStatus:"+(reservation.isPaid ?"Paid":"Pending"));
```

```
}
```

```
//Method to display all available movies void
```

```
showMovies() {
```

```
    StringBuilder details=new StringBuilder("Available Movies:\n");for(
```

```
    Movie movie: movies.values()){
```

```
        details.append("Movie:").append(movie.name).append("\n");for
```

```
        (inti= 0;i<movie.showTimes.length;i++){
```

```
            details.append("ShowTime:").append(movie.showTimes[i])
```



```

        .append(" | Available
Seats:").append(movie.availableSeats[i]).append("\n");
    }
}
statusArea.setText(details.toString());
}

//Mainmethod
public static void main(String[] args)
{
    newMovieBooking().setVisible(true);
}
}

```

## APPENDIX B - SCREENSHOTS

### SHOW AVAILABLE MOVIES

Movie Ticket Booking System

Select Movie:  Select Show Time:  Enter Number of Seats:  Enter Your Name:

Enter Reservation ID:  Payment Method:  Enter Payment Details:

Available Movies:

Movie: Jurassic Park  
Show Time: 11:00 AM | Available Seats: 50  
Show Time: 2:00 PM | Available Seats: 50  
Show Time: 5:00 PM | Available Seats: 50

Movie: Veronica  
Show Time: 12:00 PM | Available Seats: 50  
Show Time: 3:00 PM | Available Seats: 50  
Show Time: 6:00 PM | Available Seats: 50

Movie: Black  
Show Time: 10:00 AM | Available Seats: 50

### BOOK TICKETS

Movie Ticket Booking System

Select Movie:  Select Show Time:  Enter Number of Seats:  Enter Your Name:

Enter Reservation ID:  Payment Method:  Enter Payment Details:

Booking successful!  
Reservation ID: RES2  
Movie: Black  
Show Time: 4:00 PM  
Seats Booked: 2  
Amount to Pay: ₹400  
Payment Pending

## MAKE PAYMENT

Movie Ticket Booking System

Select Movie:  Select Show Time:  Enter Number of Seats:  Enter Your Name:

Enter Reservation ID:  Payment Method:  Enter Payment Details:

Payment Successful!  
Reservation ID: RES1  
Amount Paid: ₹400  
Payment Method: UPI

## VIEW RESERVATION

Movie Ticket Booking System

Select Movie:  Select Show Time:  Enter Number of Seats:  Enter Your Name:

Enter Reservation ID:  Payment Method:  Enter Payment Details:

Reservation Details:  
Reservation ID: RES1  
Movie: Black  
Show Time: 4:00 PM  
Seats Booked: 2  
Amount to Pay: ₹400  
Customer Name: Sugapriya  
Payment Status: Paid