

**Assignment 1**

**Cinema Management System**

**Submitted By:**

Muhammad Zayeem

**Submitted To:**

Shahid Bhatti

**Course Name:**

Object Oriented Programming

**COMSATS University Islamabad,**

**Lahore Campus**

**Cinema Management System**

# List Of Classes

* SeatType (Enum)
* Seat
* SeatDemo
* Screen
* ScreenDemo
* Cinema
* CityCinema

# **SEAT CLASS**

## Booking/Cancelling Method

public boolean bookSeat(){

return isAvailable=false;

}

public boolean cancelBooking(){

return isAvailable=true;

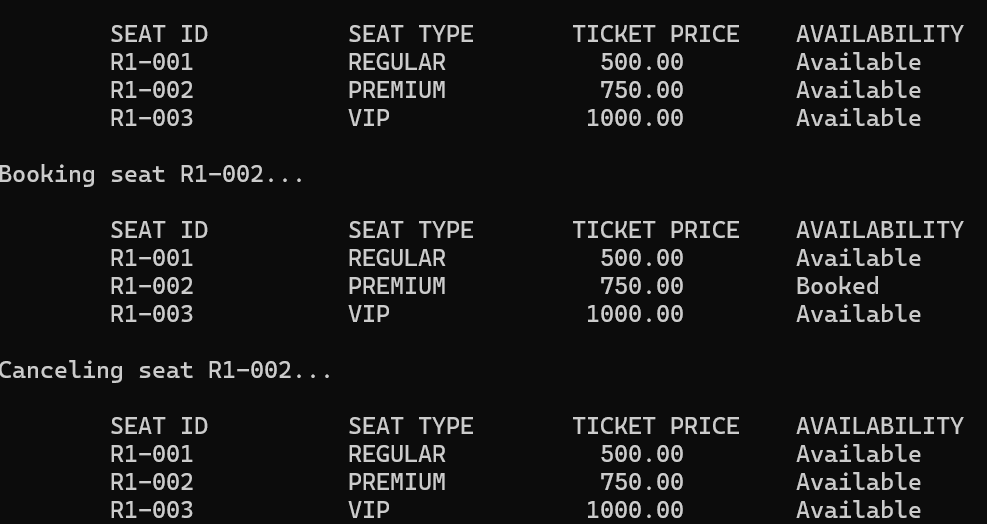
}

public String display\_availability(boolean isAvailable){

if(isAvailable==true){

return "Available";}

else

return "Booked";} 

## Availability Symbol Method.

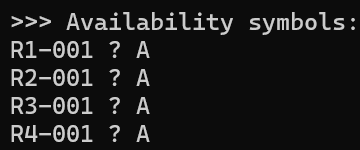
public String display\_availabilitysymbol(){

if(getisAvailable()==true){

return "A";}

else

return "B";

}

# SCREEN CLASS

## Layout Display Method

public void displaylayout(){

for(int i=0;i<seats.length;i++){

for(int j=0;j<seats[i].length;j++){

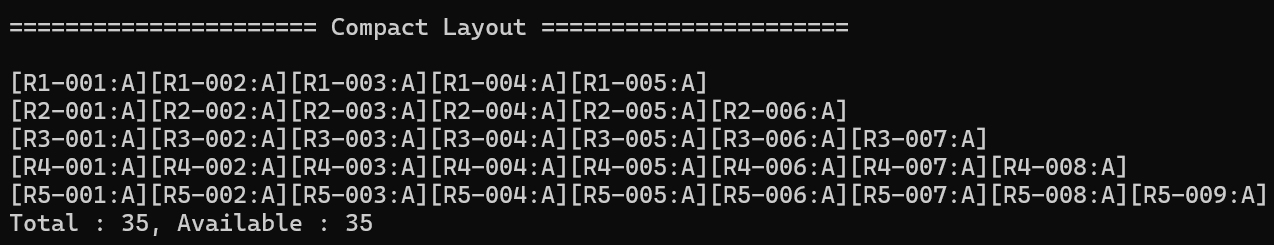
System.out.print("["+seats[i][j].getseatID()+":"+seats[i][j].display\_availabilitysymbol()+"]");

}

System.out.println();

}

System.out.println("Total : "+totalSeatCount()+", Available : "+getAvailableSeatCount());

}

## Booking Method

public boolean book(int row,int seat){

if (!checkBound(row, seat)) {

System.out.println("Invalid Input! Please try again.");

return true;

}

if(seats[row-1][seat-1].getisAvailable()){

seats[row-1][seat-1].bookSeat();

return false;

}

else{

return true;

}

}

## A screen shot of a computerCancelling Method

public boolean cancelbooking(int row,int seat){

if (!checkBound(row, seat)) {

System.out.println("Invalid Input! Please try again.");

return false;

}

if(!seats[row-1][seat-1].getisAvailable()){

seats[row-1][seat-1].cancelBooking();

return true;

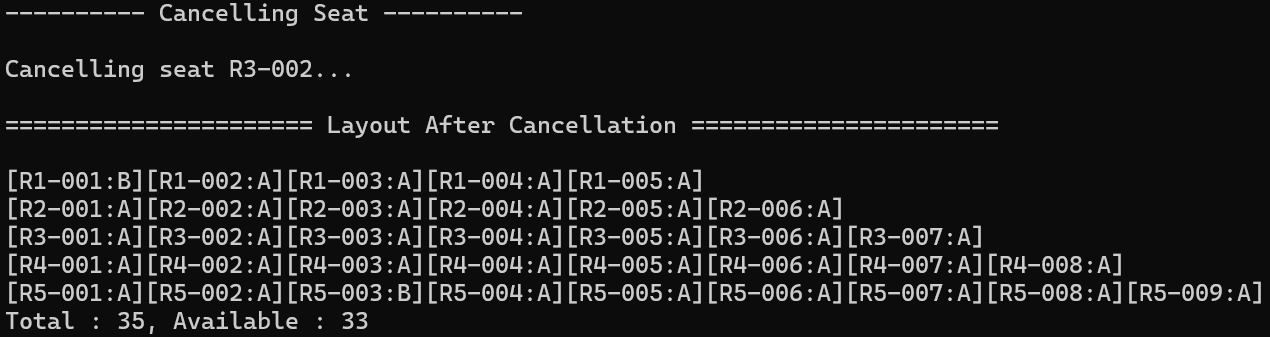
}

else{

return false;

}

}



## Available Seats by Type

public int getAvailableSeatCount(SeatType s){ //Available Seats by Seat Type.

int availableSeat=0;

for(int i=0;i<seats.length;i++){

for(int j=0;j<seats[i].length;j++){

if(seats[i][j].getisAvailable()==true && seats[i][j].gettype()==s){

availableSeat++;

}

}

}

return availableSeat;

}

A black screen with white text

AI-generated content may be incorrect.m

## First Available Seat By Type

public Seat getFirstAvailableSeat(SeatType st){ //First available seat of particular type.

for(int i=0;i<seats.length;i++){

for(int j=0;j<seats[i].length;j++){

if(seats[i][j].getisAvailable()==true && seats[i][j].gettype()==st){

System.out.println(s.display());

return seats[i][j];}}}

return null;}

## A black screen with white text AI-generated content may be incorrect.List Of Available Seat

public Seat[] getListAvailableSeat(SeatType s) {

int count = 0;

for (int i=0;i<seats.length;i++) {

for (int j=0;j<seats[i].length;j++) {

if (seats[i][j].getisAvailable() && seats[i][j].gettype() == s) {

count++;}}}

Seat[] list = new Seat[count];

int index = 0;

for (int i=0;i<seats.length;i++) {

for (int j=0;j<seats[i].length;j++) {

if (seats[i][j].getisAvailable() && seats[i][j].gettype() == s) {

list[index]=seats[i][j];

index++; }}} return list;}

## A black screen with white numbers AI-generated content may be incorrect.Changing Row Type and Price

public void setRowType(int row,SeatType s,double d){ //set seat info by row,seat-type,price

if (row<1 || row>seats.length){

System.out.println("Invalid row number!");

return;}

for(int i=0;i<seats[row-1].length;i++){

seats[row-1][i].setSeatType(s);

seats[row-1][i].setprice(d);}}

