**Assignment 2(16-20):**

**Assignment 16:**

**Question**

Assignment #16

# Player Skill

Create a class named **Player** with the following private member variables.

* String name
* String country
* Skill skill

 Use toString() method ("%-15s %-15s %-15s",to display the Player details)  
  
Create a class named Skill with the following private member variables.

* String skillName

Use toString() method ("%-15s",to display the Skill details)  
Include appropriate getters, setters and constructors.  
  
Create a class named **PlayerBO** with the following public methods.  
**Code:**

**class** Player {

**private** String name;

**private** String country;

**private** Skill skill;

//constructor

**public** Player(String name, String country, Skill skill) {

**this**.name = name;

**this**.country = country;

**this**.skill = **new** Skill();

**this**.skill = skill;

}

//getter setter

**public** String getName() {

**return** name;

}

**public** String getCountry() {

**return** country;

}

**public** Skill getSkill() {

**return** skill;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **void** setCountry(String country) {

**this**.country = country;

}

**public** **void** setSkill(Skill skill) {

**this**.skill = skill;

}

**public** String toString() {

String string;

string = (String) String.*format*("%-15s %-15s %-15s", **this**.name, **this**.country, **this**.skill.getSkillName());

**return** string;

}

}

**class** Skill {

**private** String skillName;

**public** String getSkillName() {

**return** skillName;

}

**public** **void** setSkillName(String skillName) {

**this**.skillName = skillName;

}

}

**class** PlayerBo {

**public** **void** viewDetails(Player[] playerList) {

System.***out***.format("%-15s %-15s %-15s", "Player", "Country", "Skills");

System.***out***.println();

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

System.***out***.println(playerList[i].toString());

System.***out***.println("\n");

}

}

**public** **void** PlayerDetailsWithSkill(Player[] playerList, String skill) {

**int** flag= 0;

Formatter formatter= **new** Formatter();

formatter.format("%15s %15s %15s\n", "Player", "Country", "Skill");

**for**(Player player: playerList)

{**if**((player.getSkill()).getSkillName().equals(skill))

{flag= 1;formatter.format("%15s %15s %15s\n", player.getName(),

player.getCountry() , player.getSkill().getSkillName()); }}

System.***out***.println(formatter);

**if**(flag== 0)

{System.***out***.println("Skill not found");}}

}

**package** com.cg.Player\_Skill;

**import** java.util.Scanner;

**public** **class** Main {

**public** **static** **void** main(String[] args) {

**try** (Scanner myObj = **new** Scanner(System.***in***)) {

System.***out***.println("Enter the number of players ");

**int** num = myObj.nextInt();

Player[] obj1 = **new** Player[num];

Skill[] obj2= **new** Skill[num];

PlayerBo obj3 = **new** PlayerBo();

String name, country, skill;

**for**(**int** i=0;i<num;i++) {

System.***out***.println("Enter the player " +(i+1)+ " details");

System.***out***.println("Enter the player name: ");

name = myObj.next();

System.***out***.println("Enter Country name: ");

country = myObj.next();

System.***out***.println("Enter skill name: ");

skill = myObj.next();

obj2[i] = **new** Skill();

obj2[i].setSkillName(skill);

obj1[i] = **new** Player(name, country, obj2[i]);

}

**int** str1;

**do** {

System.***out***.println("Menu: ");

System.***out***.println("1. View Details\n 2. Filter players with skill\n 3.Exit");

System.***out***.print("Enter your choice: ");

str1 = myObj.nextInt();

**switch**(str1) {

**case** 1:

obj3.viewDetails(obj1);

**break**;

**case** 2:

System.***out***.println("Enter the skill: ");

String str2 = myObj.next();

obj3.printPlayerDetailsWithSkill(obj1, str2);

**break**;

}

}

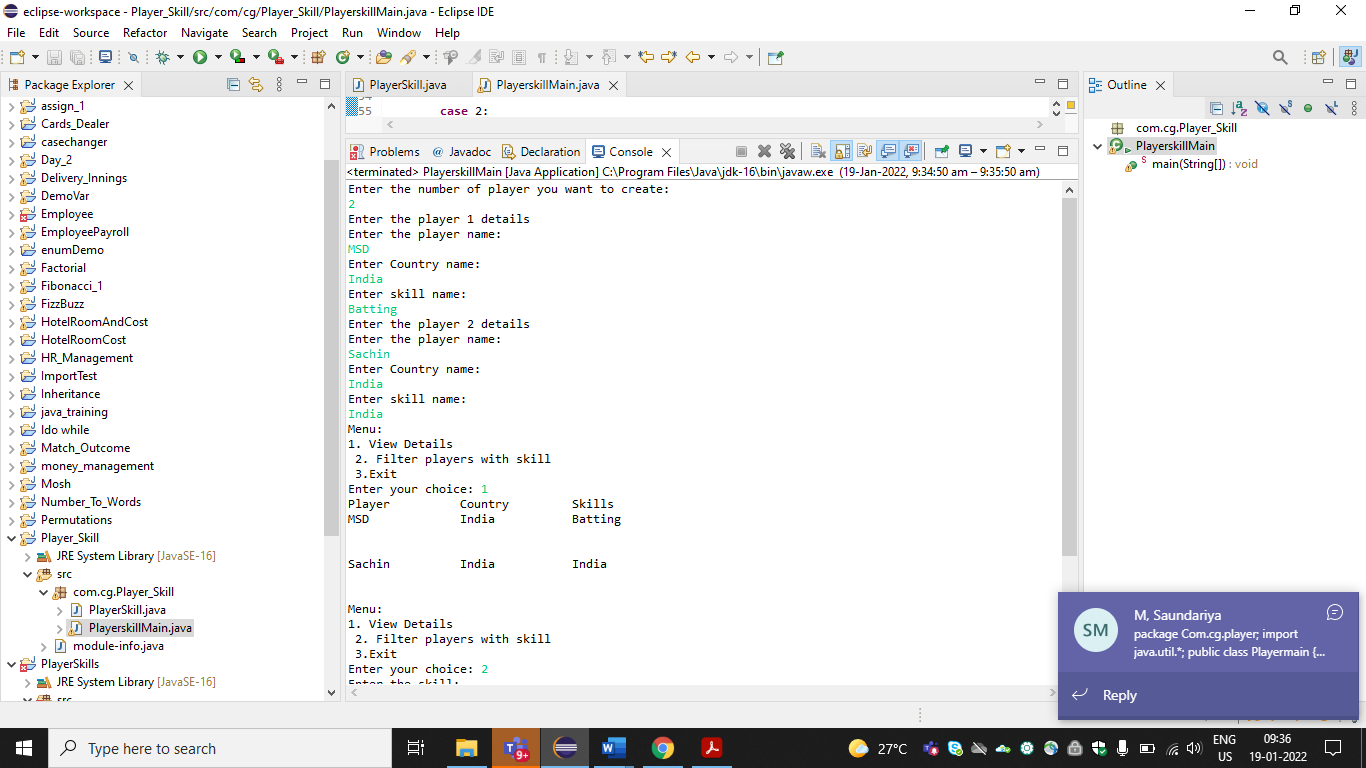
**while**(str1 != 3);

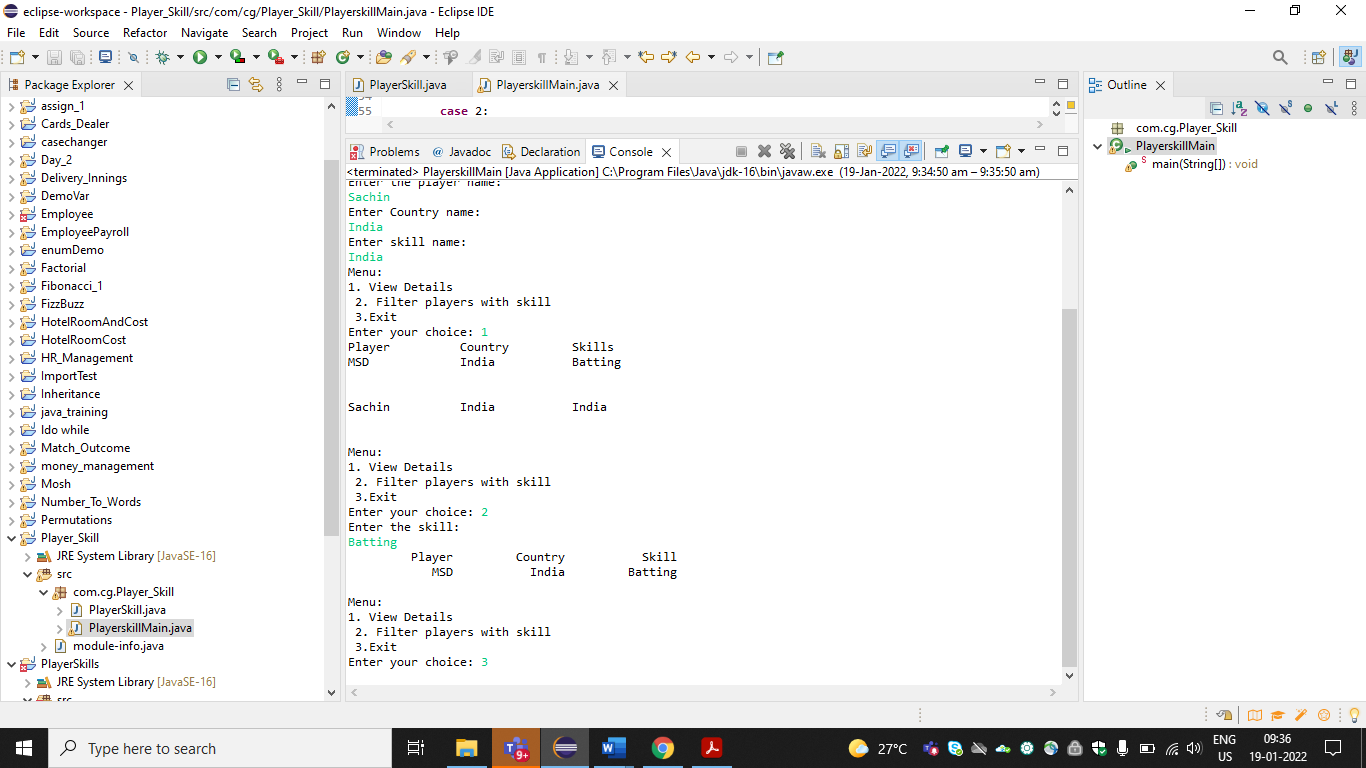
}

}

}

**Output:**





**Ouestion17:**

Assignment #17

# Team and Match Detail

Create a class named **Player** with the following private member variables / attributes.

* String  name;
* String country;
* String skill;

Include appropriate getters, setters and constructors.   
    
Create a class named **PlayerBO** and include the following methods

|  |  |  |
| --- | --- | --- |
| No | Method Name | Method Description |
| 1 | Player createPlayer (String data) | In this method, construct a player object.Split the comma seperated input. |

Create a class named  **Team** with the following private member variables / attributes.

* String  name;
* Player player;

Include appropriate getters, setters and constructors.   
  
Create a class named  **TeamBO**and include the following methods

|  |  |  |
| --- | --- | --- |
| No | Method Name | Method Description |
| 1 | public Team createTeam(String data, Player[] PlayerList) | In this method, parse the string data using Split method defined in the string class and and construct a team object. Iterate through the playerList and get the player object reference. |

Create a class named  **Match** with the following private member variables / attributes.

* String  date;
* Team teamone;
* Team teamtwo;
* String venue
* Team team

Include appropriate getters, setters and constructors.

Override the ToString() method to display the match details in the format specified in the output.   
Use this format String.format("%-15s %-15s %-15s %s", to display match details);

**Code:**

**package** com.cg.team;

**import** java.util.Scanner;

**public** **class** PlayerBo {

String player[];

String team[];

String match[];

String res;

String res1;

String[] split;

String[] split1;

String s2;

String s1;

String filter;

**int** num,num1,num2;

**void** Player()

{

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Enter player count");

num = sc.nextInt();

player = **new** String[num];

**for**(**int** i=0;i<num;i++)

{

Scanner obj = **new** Scanner(System.***in***);

System.***out***.println("Enter player"+" "+(i+1)+" "+"details");

player[i]=obj.nextLine();

}

}

**void** Team()

{

Scanner obj1 = **new** Scanner(System.***in***);

System.***out***.println("Enter team count");

num1 = obj1.nextInt();

team = **new** String[num1];

**for**(**int** j=0;j<num1;j++)

{

Scanner obj3 = **new** Scanner(System.***in***);

System.***out***.println("Enter team"+" "+(j+1)+" "+"details");

team[j]=obj3.nextLine();

}

}

**void** Match()

{

Scanner obj4 = **new** Scanner(System.***in***);

System.***out***.println("Enter match count");

num2 = obj4.nextInt();

match = **new** String[num2];

**for**(**int** k=0;k<num2;k++)

{

Scanner obj5 = **new** Scanner(System.***in***);

System.***out***.println("Enter match"+" "+(k+1)+" "+"details");

match[k]=obj5.nextLine();

}

}

**void** findTeam()

{

System.***out***.println("Menu:");

System.***out***.println("1)Find Team");

System.***out***.println("2)Find All Matches in A Specific Venue");

System.***out***.println("Type 1 or 2");

System.***out***.println("Enter your choice");

Scanner sc6 = **new** Scanner(System.***in***);

**int** a=sc6.nextInt();

**switch** (a)

{

**case** 1:

System.***out***.println("Enter Match date");

Scanner sc7 = **new** Scanner(System.***in***);

String m\_date=sc7.nextLine();

s1=m\_date.trim();

**for**(**int** d=0;d<num2;d++)

{

res = match[d];

split = res.split(",");

**for**(**int** c=0;c<split.length;c++)

{

**if**(split[c].equals(s1))

{

System.***out***.println("Team");

System.***out***.println(split[c+1]+","+split[c+2]);

}

}

}

System.***out***.println("Do you want to continue?Type Yes or No");

Scanner obj5 = **new** Scanner(System.***in***);

String ans =obj5.nextLine();

**if**(ans.equalsIgnoreCase("Yes"))

{

findMatch();

}

**else**

{

System.*exit*(0);

}

**break**;

**default**:

System.***out***.println("Invalid Input!");

}

}

**void** findMatch()

{

System.***out***.println("Menu:");

System.***out***.println("1)Find Team");

System.***out***.println("2)Find All Matches in A Specific Venue");

System.***out***.println("Type 1 or 2");

System.***out***.println("Enter your choice");

Scanner objj = **new** Scanner(System.***in***);

**int** s=objj.nextInt();

System.***out***.println("Match details");

System.***out***.println("Enter Team Name");

Scanner obj7 = **new** Scanner(System.***in***);

String name=obj7.nextLine();

s2=name.trim();

String s1=String.*format*("%-15s","Date");

String s2=String.*format*("%-15s","Teamone");

String s3=String.*format*("%-15s","Teamtwo");

String s4=String.*format*("%-15s","Venue");

System.***out***.println(s1+s2+s3+s4);

**for**(**int** e=0;e<num2;e++)

{

res1 = match[e];

split1 = res1.split(",");

**for**(**int** w=0;w<split.length;w++)

{

**if**(split1[w+1].equals(s2) || split1[w+2].equals(s2))

{

String s5=String.*format*("%-15s",split1[w]);

String s6=String.*format*("%-15s",split1[w+1]);

String s7=String.*format*("%-15s",split1[w+2]);

String s8=String.*format*("%-15s",split1[w+3]);

System.***out***.println(s5+s6+s7+s8);

}

}

}

System.***out***.println("Do you want to continue?Type Yes or No");

Scanner obj6 = **new** Scanner(System.***in***);

String ans =obj6.nextLine();

**if**(ans.equals("Yes"))

{

findTeam();

}

**else**

{

System.*exit*(0);

}

}

}

**MAIN class**

**package** com.cg.team;

**public** **class** TeamMain {

**public** **static** **void** main(String[] args)

{

PlayerBo teammatch = **new** PlayerBo();

teammatch.Player();

teammatch.Team();

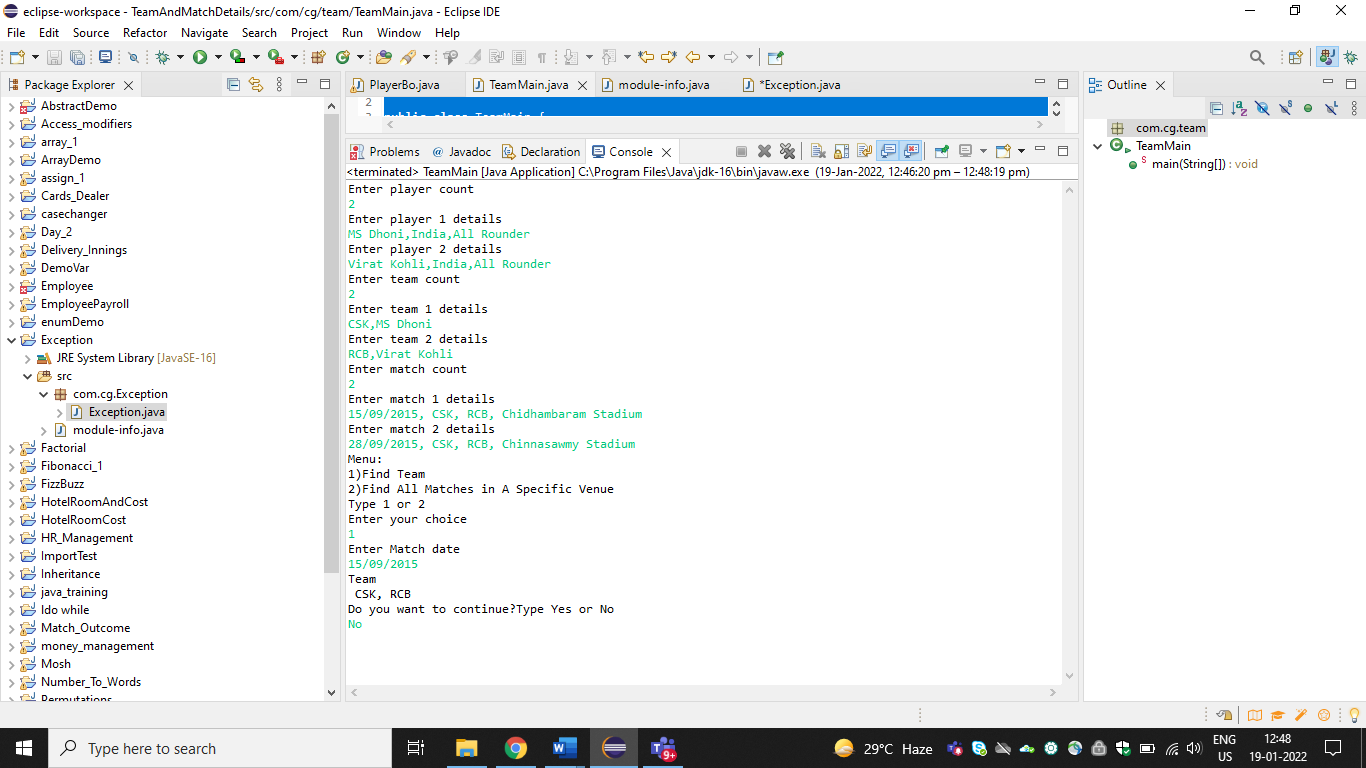
teammatch.Match();

teammatch.findTeam();

}

}

**Output:**



**Questions 18:**

# Innings-Delivery

Create a class named **Innings** with the following private member variables / attributes.

* Long inningsNumber;
* String battingTeam;

Include appropriate getters, setters and constructors.   
  
Create a class named InningsBO and include the following methods

|  |  |  |
| --- | --- | --- |
| No | Method Name | Method Description |
| 1 | Innings createInnings (String data) | In this method, parse the string data using split method defined in the String class and and construct a innings object. |

Create a class named **Delivery** with the following private member variables / attributes.

* Long deliveryNumber
* String batsman;
* String bowler;
* Long runs;
* Long inningsNumber;
* Innings innings

Create a class named **DeliveryBO** and include the following methods

|  |  |  |
| --- | --- | --- |
| No | Method Name | Method Description |
| 1 | public Delivery createDelivery(String data, Innings [] inningsList) | In this method, parse the string data using split method defined in the String class and and construct a delivery object. Iterate through the InningsList and get the Innings object reference. |
| 2 | public String findInningsNumber(Innings [] inningsList, long deliveryNumber) | In this method, find the innings to which a particular delivery belongs to. |

Create a Main class to test the above 4 classes. In the main method, create an array of Innings instances / objects and an array of delivery instances / objects. Invoke the various methods in the BO class and test them.   
**Input and Output Format:**   
Refer sample input and output for formatting specifications.   
All text in bold corresponds to input and the rest corresponds to output.   
**Code:**

**package** com.cg.Innings;

**import** java.util.Scanner;

**public** **class** InningsBo {

String delivery [];

String inning[];

String result;

String str;

**int** num1 ;**int** num2;**int** num3;

String[] split1;

**void** inning()

{

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Enter number of innings");

num1 = sc.nextInt();

inning = **new** String[num1];

**for**(**int** i=0;i<num1;i++)

{

Scanner obj = **new** Scanner(System.***in***);

System.***out***.println("Enter innings"+" "+(i+1)+" "+"details");

inning[i]=obj.nextLine();

}

}

**void** delivery()

{

Scanner obj1 = **new** Scanner(System.***in***);

System.***out***.println("Enter number of deliveries");

num2 = obj1.nextInt();

delivery = **new** String[num2];

**for**(**int** j=0;j<num2;j++)

{

Scanner obj2 = **new** Scanner(System.***in***);

System.***out***.println("Enter inning"+" "+(j+1)+" "+"details");

delivery[j]=obj2.nextLine();

}

}

**void** result()

{

Scanner obj3 = **new** Scanner(System.***in***);

System.***out***.println("Enter the delivery number for which you need to find the innings number");

num3 = obj3.nextInt();

str=String.*valueOf*(num3);

**for**(**int** k=0;k<num2;k++)

{

result = delivery[k];

split1 = result.split(",");

**for**(**int** c=0;c<split1.length;c++)

{

**if**(split1[c].equals(str))

{

System.***out***.println("Innings:"+split1[c+4]);

}

}

}

}

}

**class** DeliveryBo

{

**public** **static** **void** main(String[] args)

{

InningsBo inningdelivery = **new** InningsBo();

inningdelivery.inning();

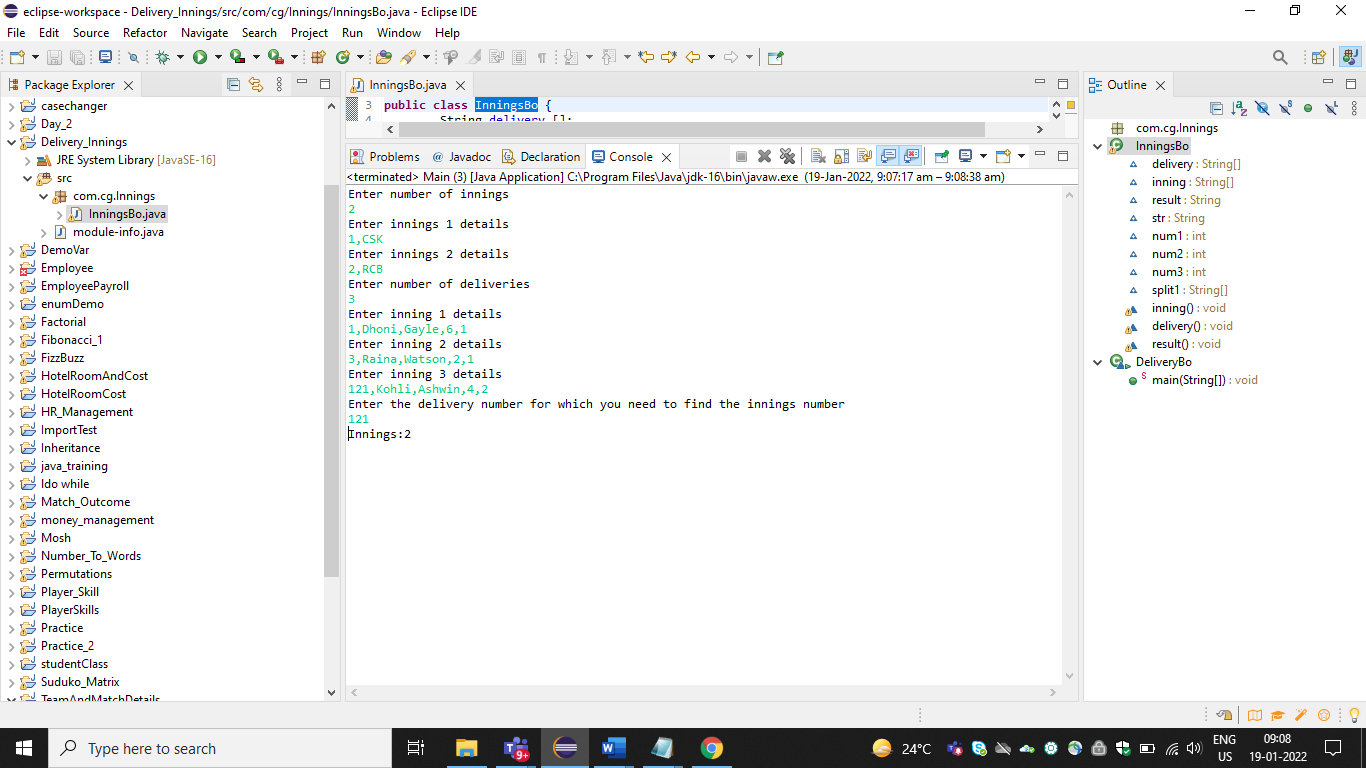
inningdelivery.delivery();

inningdelivery.result();

}

}

**Output:**



**Question 19:**

# Match-Outcome

Create a class named **Outcome** with the following private member variables.

|  |  |
| --- | --- |
| **Data Type** | **Variable Name** |
| String | status |
| String | winnerTeam |

Include appropriate getters, setters and constructors.   
Naming Conversion:   
Getter: getStatus, getWinnerTeam   
Setter: setStatus, setWinnerTeam   
Include 2 argument constructor. The first argument corresponds to status and the second argument corresponds to winnerTeam.   
Override the toString() method to display the match details in the format specified in the output.   
**String.format("%-15s %-15s", status, winnerTeam);**  
Create a class named Match with the following private member variables.

|  |  |
| --- | --- |
| Data Type | Variable Name |
| String | date |
| String | teamOne |
| String | teamTwo |
| String | venue |
| Outcome | outcome |

Include appropriate getters, setters and constructors.   
Naming Conversion:   
Getter: getDate,getTeamOne,etc..   
Setter: setDate,setTeamOne,etc..   
Include 5 argument constructor. The order in which the arguments are passed is date,teamOne,teamTwo,venue,outcome.   
  
Override the toString() method to display the match details in the format specified in the output.   
**String.format("%-15s %-15s %-15s %-15s %-30s",date,teamOne,teamTwo,venue,outcome);**  
**Code:**

**package** com.cg.Match;

**import** java.util.Scanner;

**public** **class** Outcome {

String status[];

String winnerTeam[];

String matchDate[];

String teamOne[];

String teamTwo[];

String venue[];

String outComeStatus;

String outComeWinnerTeam;

String string\_1;

String string2;

String result1;

String result2;

**int** number ;

**void** Match\_1()

{

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Enter number of matches");

number = sc.nextInt();

matchDate = **new** String[number];

teamOne = **new** String[number];

teamTwo = **new** String[number];

venue = **new** String[number];

status = **new** String[number];

winnerTeam = **new** String[number];

**for**(**int** i=0;i<number;i++)

{

Scanner obj = **new** Scanner(System.***in***);

System.***out***.println("Enter match"+" "+(i+1)+" "+"details");

System.***out***.println("Enter match date");

matchDate[i]=obj.nextLine();

System.***out***.println("Enter team one");

teamOne[i]=obj.nextLine();

System.***out***.println("Enter team two");

teamTwo[i]=obj.nextLine();

System.***out***.println("Enter venue");

venue[i]=obj.nextLine();

System.***out***.println("Enter status");

status[i]=obj.nextLine();

System.***out***.println("Enter winner team");

winnerTeam[i]=obj.nextLine();

}

}

**void** MatchDetails\_1()

{

System.***out***.println("Enter outcome status");

Scanner sc3 = **new** Scanner(System.***in***);

outComeStatus=sc3.nextLine();

string\_1=outComeStatus.trim();

System.***out***.println("Match Details");

String data1=String.*format*("%-15s","Date");

String data2=String.*format*("%-15s","Team1");

String data3=String.*format*("%-15s","Team2");

String data4=String.*format*("%-15s","Venue");

String data5=String.*format*("%-15s","Status");

String data6=String.*format*("%-15s","Winner");

System.***out***.println(data1+data2+data3+data4+data5+data6);

**for**(**int** j=0;j<status.length;j++)

{

**if**(status[j].equals(string\_1))

{

String data7=String.*format*("%-15s",matchDate[j]);

String data8=String.*format*("%-15s",teamOne[j]);

String data9=String.*format*("%-15s",teamTwo[j]);

String data10=String.*format*("%-15s",venue[j]);

String data11=String.*format*("%-15s",status[j]);

String data12=String.*format*("%-15s",winnerTeam[j]);

System.***out***.println(data7+data8+data9+data10+data11+data12);

}

}

}

**void** MatchDetailsWithOutcomeWinnerTeam()

{

System.***out***.println("Enter outcome winner team");

Scanner myObj = **new** Scanner(System.***in***);

outComeWinnerTeam=myObj.nextLine();

string2=outComeWinnerTeam.trim();

System.***out***.println("Match Details");

String str1=String.*format*("%-15s","Date");

String str2=String.*format*("%-15s","Team1");

String str3=String.*format*("%-15s","Team2");

String str4=String.*format*("%-15s","Venue");

String str5=String.*format*("%-15s","Status");

String str6=String.*format*("%-15s","Winner");

System.***out***.println(str1+str2+str3+str4+str5+str6);

**for**(**int** k=0;k<winnerTeam.length;k++)

{

**if**(winnerTeam[k].equals(string2))

{

String str7=String.*format*("%-15s",matchDate[k]);

String str8=String.*format*("%-15s",teamOne[k]);

String str9=String.*format*("%-15s",teamTwo[k]);

String str10=String.*format*("%-15s",venue[k]);

String str11=String.*format*("%-15s",status[k]);

String str12=String.*format*("%-15s",winnerTeam[k]);

System.***out***.println(str7+str8+str9+str10+str11+str12);

}

}

}

**void** matchDetails()

{

System.***out***.println("Match Details");

String obj1=String.*format*("%-15s","Date");

String obj2=String.*format*("%-15s","Team One");

String obj3=String.*format*("%-15s","Team Two");

String obj4=String.*format*("%-15s","Venue");

String obj5=String.*format*("%-15s","Status");

String obj6=String.*format*("%-15s","Winner");

System.***out***.println(obj1+obj2+obj3+obj4+obj5+obj6);

**for**(**int** j=0;j<number;j++)

{

String obj7=String.*format*("%-15s",matchDate[j]);

String obj8=String.*format*("%-15s",teamOne[j]);

String obj9=String.*format*("%-15s",teamTwo[j]);

String obj10=String.*format*("%-15s",venue[j]);

String obj11=String.*format*("%-15s",status[j]);

String obj12=String.*format*("%-15s",winnerTeam[j]);

System.***out***.println(obj7+obj8+obj9+obj10);

}

}

}

**package** com.cg.Match;

**import** java.util.Scanner;

**public** **class** MactchBo {

**public** **static** **void** main(String[] args)

{

Outcome matchoutcome = **new** Outcome();

matchoutcome.Match\_1();

**for**(**int** i=0;i<4;i++)

{

System.***out***.println("Menu:");

System.***out***.println("1.View match details");

System.***out***.println("2.Filter match details with outcome status");

System.***out***.println("3.Filter match details with outcome winner team");

System.***out***.println("4.Exit");

System.***out***.println("Enter your choice");

Scanner myObj1 = **new** Scanner(System.***in***);

**int** n=myObj1.nextInt();

**switch**(n)

{

**case** 1:

matchoutcome.matchDetails();

**break**;

**case** 2:

matchoutcome. MatchDetails\_1();

**break**;

**case** 3:

matchoutcome.MatchDetailsWithOutcomeWinnerTeam();

**break**;

**case** 4:

System.*exit*(0);

**break**;

**default**:

System.***out***.println("Invalid Input!");

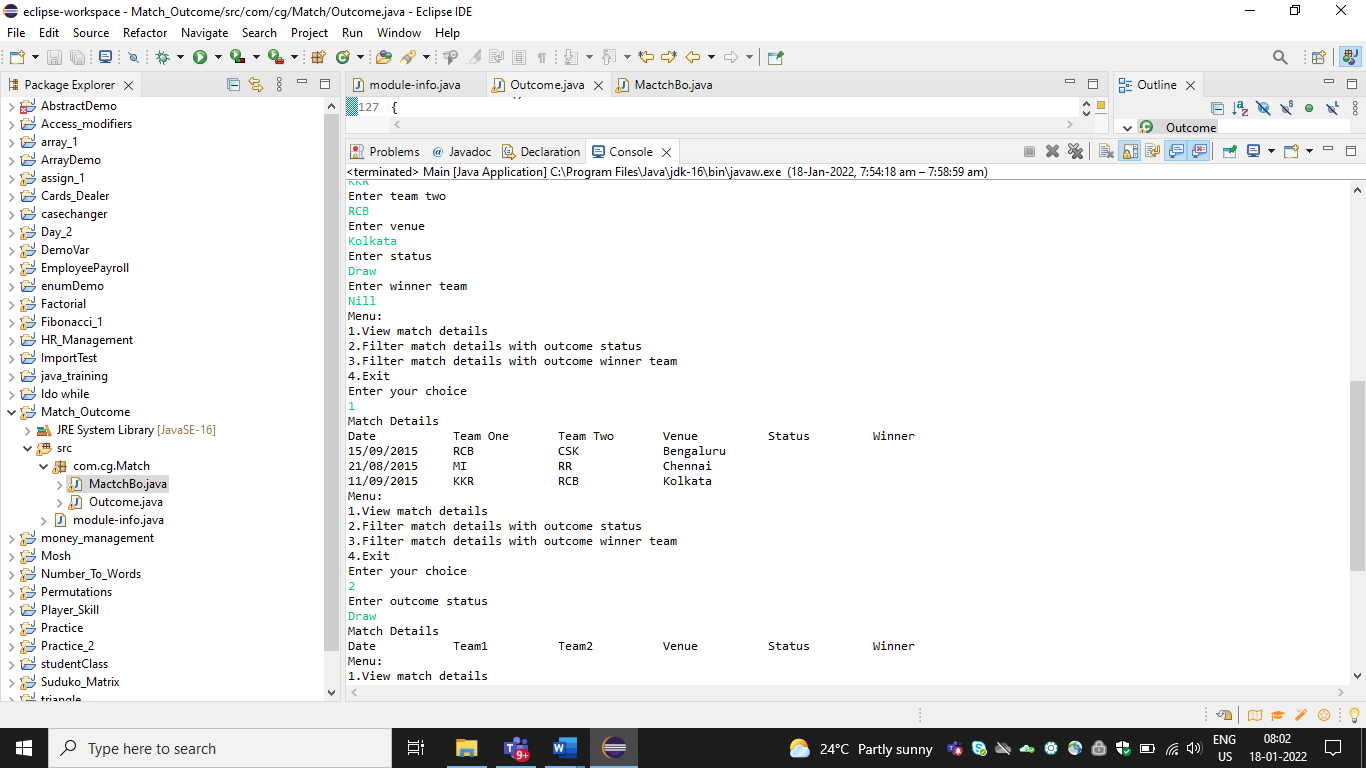
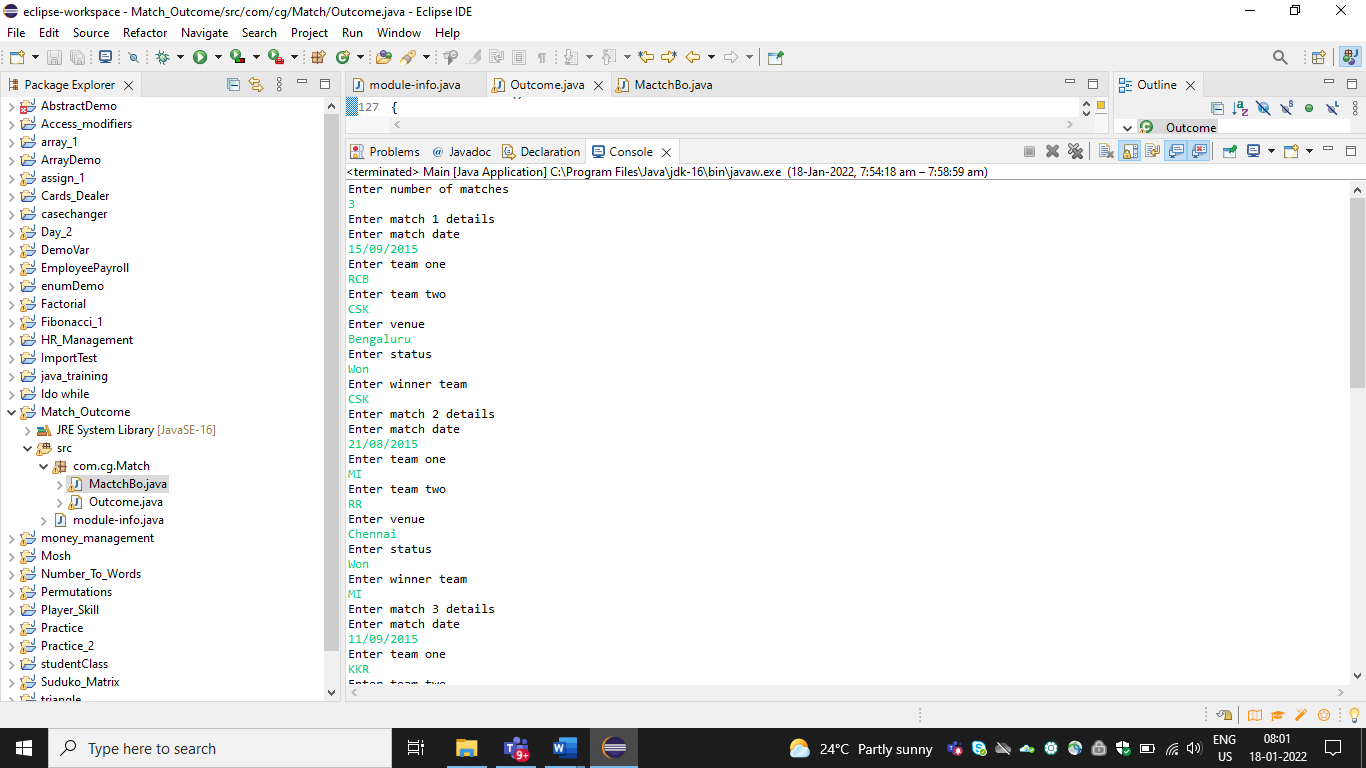
}

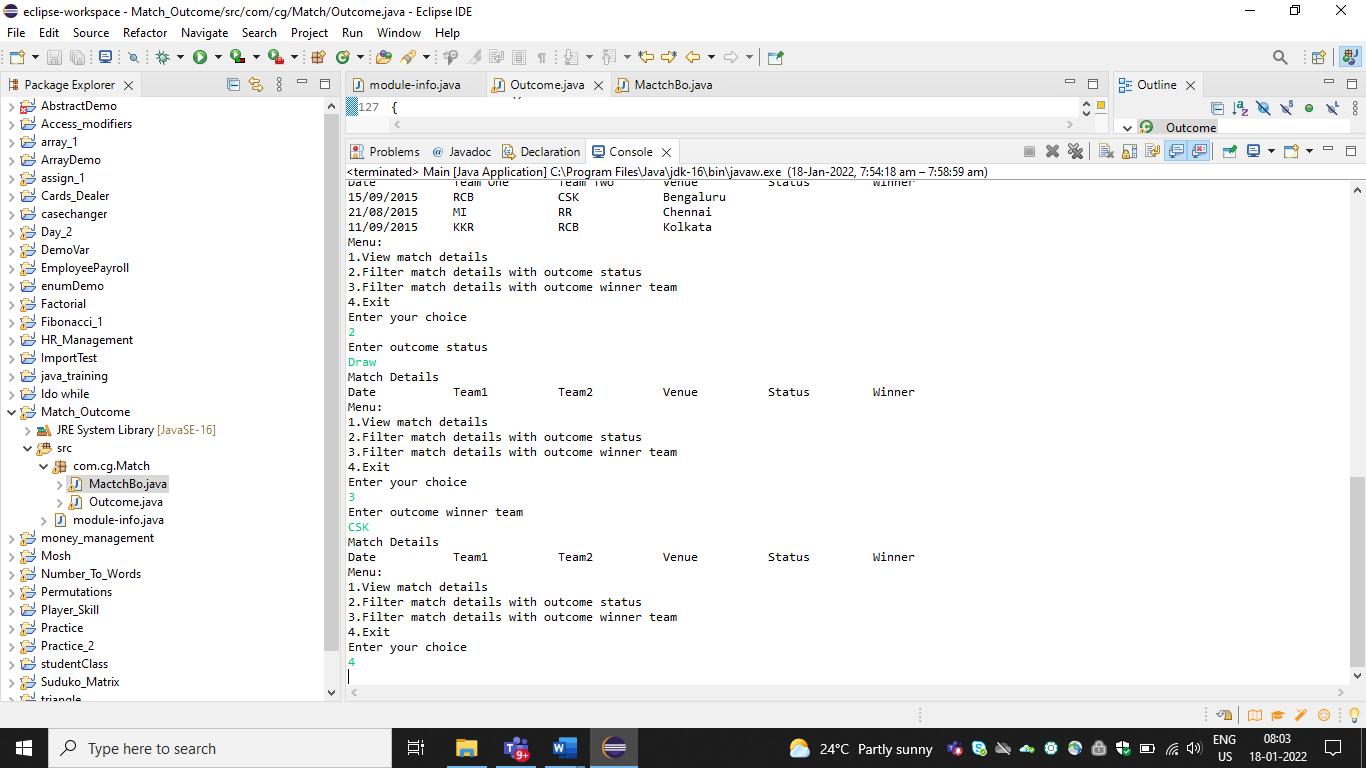
}

}

}

**Output:**





**Question 20:**

# Hotel Room and Cost

The task is to get the details of the hotel room and display the final cost of the room using a menu driven application.

Write a Java program to implement this task.

 Create a class **HotelRoom**

Include the following protected data members / attributes:

* hotelName – of type String
* numberOfSqFeet – of type Integer
* hasTV - of type Boolean
* hasWifi – of type Boolean

Include the following public methods :

* Create a constructor that initializes all the data members
* public HotelRoom(String hotelName,Integer numberOfSqFeet,Boolean hasTV,Boolean hasWifi)
* "calculateTariff" – Calculates cost using the number of sq feets and cost per sq feet and returns an Integer
* "getRatePerSqFeet" - This method returns an Integer. In this case, it always returns 0

Create a class **DeluxeRoom** that extends **HotelRoom**

Include the following protected attributes / data members:

* ratePerSqFeet – of type Integer.

Include the following public methods :

* Include a constructor that sets ratePerSqFeet as 10.
* public HotelRoom(String hotelName,Integer numberOfSqFeet,Boolean hasTV,Boolean hasWifi)
* "getRatePerSqFeet" – returns (ratePerSqFeet + 2) if wifi is present, else returns ratePerSqFeet.

Create a class **DeluxeACRoom** that extends DeluxeRoom

Include the following public methods:

* Include a constructor that sets ratePerSqFeet as 12.

Create a class **SuiteACRoom** that extends HotelRoom

Include the following private attributes / data members:

* ratePerSqFeet – of type Integer.

Include the following public methods :

* Include a constructor that sets ratePerSqFeet as 15.
* public HotelRoom(String hotelName,Integer numberOfSqFeet,Boolean hasTV,Boolean hasWifi)
* getRatePerSqFeet – returns (ratePerSqFeet + 2) if wifi is present, else returns ratePerSqFeet.

**Code:**

**package** com.cg.hotelroom;

**import** java.util.\*;

**public** **class** Hotel\_Room {

**public** **static** **void** main(String[] args) {

System.***out***.println("Hotel Tarrif Calculator");

System.***out***.println(" 1.Deluxe Room");

System.***out***.println(" 2.Deluxe AC Room");

System.***out***.println(" 3.Suite AC Room");

@SuppressWarnings("resource")

Scanner myObj = **new** Scanner(System.***in***);

**int** roomType = 0;

String hotelName;

Integer noOfSqFt;

System.***out***.println("Select Room Type:");

roomType = myObj.nextInt();

System.***out***.println("Hotel Name:");

hotelName = myObj.next();

System.***out***.println("Room Square Feet Area:");

noOfSqFt = myObj.nextInt();

System.***out***.println("Room has TV (yes/no):");

String TV = myObj.next();

Boolean hasTV = **false**;

**if** (TV.equalsIgnoreCase("yes")) {

hasTV = **true**;

}

System.***out***.println("Room has Wifi (yes/no):");

String isWifi = myObj.next();

Boolean WifiAvailable = **null**;

**if** (isWifi.equalsIgnoreCase("yes")) {

WifiAvailable = **true**;

}

Integer ratePerSqFeet = **null**;

Hotel\_Room hotelRoomAndCost = **new** Hotel\_Room();

DeluxeRoom deluxeRoom = hotelRoomAndCost.**new** DeluxeRoom(hotelName, noOfSqFt, hasTV, WifiAvailable, ratePerSqFeet);

DeluxeACRoom deluxeACRoom = hotelRoomAndCost.**new** DeluxeACRoom(hotelName, noOfSqFt, hasTV, WifiAvailable, ratePerSqFeet);

SuiteACRoom suiteACRoom = hotelRoomAndCost.**new** SuiteACRoom(hotelName, noOfSqFt, hasTV, WifiAvailable);

**if** (roomType == 1) {

System.***out***.println("Room Tariff per day is: " + deluxeRoom.rate);

}

**if** (roomType == 2) {

System.***out***.println("Room Tariff per day is: " + deluxeACRoom.rate);

}

**if** (roomType == 3) {

System.***out***.println("Room Tariff per day is: " + suiteACRoom.rate);

}

}

**class** HotelRoom {

**protected** String hotelName;

**protected** Integer numberOfSqFeet;

**protected** Boolean hasTV;

**protected** Boolean hasWifi;

**public** String getHotelName() {

**return** hotelName;

}

**public** **void** setHotelName(String hotelName) {

**this**.hotelName = hotelName;

}

**public** Integer getNumberOfSqFeet() {

**return** numberOfSqFeet;

}

**public** **void** setNumberOfSqFeet(Integer numberOfSqFeet) {

**this**.numberOfSqFeet = numberOfSqFeet;

}

**public** Boolean getHasTV() {

**return** hasTV;

}

**public** **void** setHasTV(Boolean hasTV) {

**this**.hasTV = hasTV;

}

**public** Boolean getHasWifi() {

**return** hasWifi;

}

**public** **void** setHasWifi(Boolean hasWifi) {

**this**.hasWifi = hasWifi;

}

**public** HotelRoom(String hotelName, Integer numberOfSqFeet, Boolean hasTV, Boolean hasWifi) {

**super**();

**this**.hotelName = hotelName;

**this**.numberOfSqFeet = numberOfSqFeet;

**this**.hasTV = hasTV;

**this**.hasWifi = hasWifi;

}

**public** **int** calculateTariff() {

**return** **this**.numberOfSqFeet \* getRatePerSqFeet();

}

**public** **int** getRatePerSqFeet() {

**return** 0;

}

}

**class** DeluxeRoom **extends** HotelRoom {

**protected** Integer ratePerSqFeet;

**public** DeluxeRoom(String hotelName, Integer numberOfSqFeet, Boolean hasTV, Boolean hasWifi,

Integer ratePerSqFeet) {

**super**(hotelName, numberOfSqFeet, hasTV, hasWifi);

}

**public** **int** getRatePerSqFeet() {

**this**.ratePerSqFeet = 10;

**if** (hasWifi) {

**return** ratePerSqFeet + 2;

} **else** {

**return** ratePerSqFeet;

}

}

**int** rate = **this**.calculateTariff();

}

**class** DeluxeACRoom **extends** DeluxeRoom {

**public** DeluxeACRoom(String hotelName, Integer numberOfSqFeet, Boolean hasTV, Boolean hasWifi,

Integer ratePerSqFeet) {

**super**(hotelName, numberOfSqFeet, hasTV, hasWifi, ratePerSqFeet);

ratePerSqFeet = 12;

}

**int** rate = **this**.calculateTariff();

}

**class** SuiteACRoom **extends** HotelRoom {

**private** Integer ratePerSqFeet;

**public** SuiteACRoom(String hotelName, Integer numberOfSqFeet, Boolean hasTV, Boolean hasWifi) {

**super**(hotelName, numberOfSqFeet, hasTV, hasWifi);

}

**public** **int** getRatePerSqFeet() {

**this**.ratePerSqFeet = 15;

**if** (hasWifi) {

**return** ratePerSqFeet + 2;

} **else** {

**return** ratePerSqFeet;

}

}

**int** rate = **this**.calculateTariff();

}

}

**Output:**

