Highest Placement

SRV college wants to recognize the department which has succeeded in getting the maximum number of placements for this academic year. The departments that have participated in the recruitment drive are CSE,ECE, MECH. Help the college find the department getting maximum placements. Check for all the possible output given in the sample snapshot

Note : If any input is negative, the output should be "Input is Invalid". If all department has equal number of placements, the output should be "None of the department has got the highest placement".

Sample Input 1:

```
Enter the no of students placed in CSE:90
Enter the no of students placed in ECE:45
Enter the no of students placed in MECH:70
```

Sample Output 1:

```
Highest placement
CSE
```

Sample Input 2:

```
Enter the no of students placed in CSE:55
Enter the no of students placed in ECE:85
Enter the no of students placed in MECH:85
```

Sample Output 2:

```
Highest placement
ECE
MECH
```

Sample Input 3:

```
Enter the no of students placed in CSE:0
Enter the no of students placed in ECE:0
Enter the no of students placed in MECH:0
```

Sample Output 3:

```
None of the department has got the highest placement
```

Sample Input 4:

```
Enter the no of students placed in CSE:10
Enter the no of students placed in ECE:-50
Enter the no of students placed in MECH:40
```

Sample Output 3:

```
Input is Invalid
```

Placement.java

```
import java.util.*;
public class Placement
{
    public static void main (String[] args) {
        Scanner sc=new Scanner(System.in);
        String[] dept=new String[]{"CSE","ECE","MECH"};
        ArrayList <Integer> placed=new ArrayList<Integer>();
        System.out.println("Enter the no of students placed in CSE:");
        int cse=sc.nextInt();
        placed.add(cse);
        System.out.println("Enter the no of students placed in ECE:");
        int ece=sc.nextInt();
        placed.add(ece);
        System.out.println("Enter the no of students placed in MECH:");
        int mech=sc.nextInt();
        placed.add(mech);
        int max=0;
        int i;
        if(cse==ece  && ece==mech)
        System.out.println("None of the department has got the highest placement");
        else if(cse<0 || ece<0 || mech<0)
        System.out.println("Input is Invalid");
        else
        {
        System.out.println("Highest placement");
        for(i=0;i<3;i++)
        {
            if(placed.get(i)>max)
            {
                max=placed.get(i);

            }
        }
        for(i=0;i<3;i++)
        {
            if(placed.get(i)==max)
            System.out.println(dept[i]);
        }
        }

    }
}
```

Contact Details of Hosteller

SNMR College of Engineering and Technology wants to create an application to store their students details as well as the details of hostellers.

In case of any changes to be made to the attributes, admin can update the details like room number and phone number of the hosteler.

Develop a program to implement this scenario.

Create a public class Student with protected attributes :

int studentId

String name

int departmentId

String gender

String phone

Create a public class Hosteller with private attributes

String hostelName

int roomNumber

Make this class inherit the Student class, as it holds all the properties of Student.

Use appropriate public getters and setters for both the classes.

Write a class Main with the main function.

In Main class get the input of the hosteller using the method :

public static Hosteller getHostellerDetails().

Invoke this method from the main method and then modify the room number and phone number, if needed.

Sample Input 1:

```
Enter the Details:
Student Id
1
Student Name
John
Department Id
101
Gender
Male
Phone Number
9876543210
Hostel Name
YMCA
Room Number
10
Modify Room Number(Y/N)
Y
New Room Number
11
Modify Phone Number(Y/N)
Y
New Phone Number
9876543121
```

Sample Output 1:

The Student Details
1 John 101 Male 9876543121 YMCA 11

Sample Input 2:
Enter the Details:
Student Id
2
Student Name
John Paul
Department Id
112
Gender
Male
Phone Number
9885526536
Hostel Name
YMBA
Room Number
5
Modify Room Number(Y/N)
N
Modify Phone Number(Y/N)
N

Sample Output 2:
The Student Details:
2 John Paul 112 Male 9885526536 YMBA 5

Student.java

```java
public class Student
{
    protected int studentId;
    protected String name;
    protected int departmentId;
    protected String gender;
    protected String phone;

        public int getStudentId() {
                return studentId;
        }
        public void setStudentId(int studentId) {
                this.studentId = studentId;
        }
        public String getName() {
                return name;
        }
        public void setName(String name) {
                this.name = name;
        }
        public int getDepartmentId() {
                return departmentId;
        }
        public void setDepartmentId(int departmentId) {
                this.departmentId = departmentId;
        }
        public String getGender() {
                return gender;
        }
        public void setGender(String gender) {
                this.gender = gender;
        }
        public String getPhone() {
                return phone;
        }
        public void setPhone(String phone) {
                this.phone = phone;
        }


}
```

Hosteller.java

```
public class Hosteller extends Student
{
    private String hostelName;
    private int roomNumber;
        public String getHostelName() {
                return hostelName;
        }
        public void setHostelName(String hostelName) {
                this.hostelName = hostelName;
        }
        public int getRoomNumber() {
                return roomNumber;
        }
        public void setRoomNumber(int roomNumber) {
                this.roomNumber = roomNumber;
        }


}
```

Main.java

```
import java.util.*;
public class Main
{
    public static Hosteller getHostellerDetails()
    {
        Hosteller h1=new Hosteller();
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the Details:");
        System.out.println("Student Id");
         int sid=sc.nextInt();
         System.out.println("Student Name");
         String sname=sc.nextLine();
         System.out.println("Department Id");
         int did=sc.nextInt();
         System.out.println("Gender");
         String gen=sc.next();
         System.out.println("Phone Number");
         String ph=sc.next();
         System.out.println("Hostel Name");
         String hname=sc.nextLine();
         System.out.println("Room Number");
         int rno=sc.nextInt();
        h1.setStudentId(sid);
        h1.setName(sname);
        h1.setDepartmentId(did);
        h1.setGender(gen);
        h1.setPhone(ph);
        h1.setHostelName(hname);
        h1.setRoomNumber(rno);
        return h1;

    }
    public static void main (String[] args) {
        Scanner sc=new Scanner(System.in);
        Hosteller h=getHostellerDetails();
        System.out.println("Modify Room Number(Y/N)");
        char ans1=sc.next().charAt(0);
        if(ans1=='Y' || ans1=='y')
        {
            System.out.println("New Room Number");
            int nrno=sc.nextInt();
            h.setRoomNumber(nrno);
        }
        System.out.println("Modify Phone Number(Y/N)");
        char ans2=sc.next().charAt(0);
        if(ans1=='Y' || ans1=='y')
        {
            System.out.println("New Phone Number");
            String phone=sc.next();
            h.setPhone(phone);
        }
        System.out.println("The Student Details");
        System.out.println(h.getStudentId()+" "+h.getName()+" "+h.getDepartmentId()+" "+h.getGender()+" "+
        h.getPhone()+" "+h.getHostelName()+" "+h.getRoomNumber());



    }
}
```

Account Manipulation - Abstract class

Yzee bank needs to automate the bank transactions. There are many accounts, like Savings Account, Current Account, Demat Account and so on.

As start up, they need to automate the Savings Account details.

You are provided with a public class Customer with private attributes :

int customerId

String customerName

String emailId

Appropriate public getters and setters are already written.

Write a public 3 argument constructor with arguments – customerId, customerName and emailId.

Write a public class Account with protected attributes :

int accountNumber

Customer customerObj

double balance

Uncomment the public getters and setters provided in the template.

Write a public 3 argument constructor with arguments – accountNumber, customerObj and balance.

Write a public method in Account class as,

public boolean withdraw(double amount) – Make this method as abstract.

Write a public class SavingsAccount with private attribute :

double minimumBalance

Uncomment the public getters and setters provided in the template.

Make this class SavingsAccount to inherit the Account class.

Write a public 4 argument constructor with arguments – accountNumber, customerObj, balance and minimumBalance.

Implement the withdraw method as

public boolean withdraw(double amount) - This method should return true if withdraw is successful, else return false.

In this method, check if

balance - amount is greater than the minimum balance.

If yes, perform withdraw. Reduce the withdraw amount from the balance and return true.

If not, return false.

Create a public class Main which has the main method. Check the correctness of the methods written in these classes.

Note : All class, methods needs to be declared as public

Customer.java

```

public class Customer {


        //Attributes
        private int  customerId;
        private String customerName;
        private String emailId;
```

```java

        //Constructor
        public Customer(int customerId, String customerName, String emailId) {

                this.customerId = customerId;
                this.customerName = customerName;
                this.emailId = emailId;
        }

    //Getters and Setters
        public int getCustomerId() {
                return customerId;
        }

        public void setCustomerId(int customerId) {
                this.customerId = customerId;
        }

        public String getCustomerName() {
                return customerName;
        }

        public void setCustomerName(String customerName) {
                this.customerName = customerName;
        }

        public String getEmailId() {
                return emailId;
        }

        public void setEmailId(String emailId) {
                this.emailId = emailId;
        }

}

```

Account.java

```java
public abstract class Account  {

  protected int accountNumber;
  protected  Customer customerObj;
  protected double balance;
  public Account(int accno,Customer CustomerObj,double balance)

{
  this.accountNumber=accno;
   this.customerObj=CustomerObj;
   this.balance=balance;
}

   public int getAccountNumber() {
                return accountNumber;
        }

        public void setAccountNumber(int accountNumber) {
                this.accountNumber = accountNumber;
        }

        public Customer getCustomerObj() {
                return customerObj;
        }

        public void setCustomerObj(Customer customerObj) {
                this.customerObj = customerObj;
        }

        public double getBalance() {
                return balance;
        }

        public void setBalance(double balance) {
                this.balance = balance;
        }
    abstract public boolean withdraw(double amount);



}
```

SavingsAccount.java

```
public class SavingsAccount extends Account {
   public SavingsAccount(int acntno,Customer cobj,double bal,double minbal)
   {   super(acntno,cobj,bal);
            this.minimumBalance=minbal;
   }


private double minimumBalance;
      public double getMinimumBalance() {
                return minimumBalance;
        }

        public void setMinimumBalance(double minimumBalance) {
                this.minimumBalance = minimumBalance;
        }
    public boolean withdraw(double amount)
    {
        if(balance-amount>minimumBalance )
        {balance=balance-amount;
        return true;
        }

           return false;

    }


}


```

Main.java

```
import java.util.*;
public class Main{

   public static void main(String args[]){
       Customer c=new Customer(123,"abc","jdhf");
      c.setCustomerId(123);
      c.setCustomerName("abc");
      c.setEmailId("jegf");
      System.out.println(c.getCustomerId());
       System.out.println(c.getCustomerName());
       System.out.println(c.getEmailId());


    }
}
```