## **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

1 *import* java.util.\*;

2 *public* *class* Placement

3 {

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

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

6 String[] dept=*new* String[]{"CSE","ECE","MECH"};

7 ArrayList <Integer> placed=*new* ArrayList<Integer>();

8 System.out.println("Enter the no of students placed in CSE:");

9 *int* cse=sc.nextInt();

10 placed.add(cse);

11 System.out.println("Enter the no of students placed in ECE:");

12 *int* ece=sc.nextInt();

13 placed.add(ece);

14 System.out.println("Enter the no of students placed in MECH:");

15 *int* mech=sc.nextInt();

16 placed.add(mech);

17 *int* max=0;

18 *int* i;

19 *if*(cse==ece && ece==mech)

20 System.out.println("None of the department has got the highest placement");

21 *else* *if*(cse<0 || ece<0 || mech<0)

22 System.out.println("Input is Invalid");

23 *else*

24 {

25 System.out.println("Highest placement");

26 *for*(i=0;i<3;i++)

27 {

28 *if*(placed.get(i)>max)

29 {

30 max=placed.get(i);

31

32 }

33 }

34 *for*(i=0;i<3;i++)

35 {

36 *if*(placed.get(i)==max)

37 System.out.println(dept[i]);

38 }

39 }

40

41 }

42 }

## **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

1 *public* *class* Student

2 {

3 *protected* *int* studentId;

4 *protected* String name;

5 *protected* *int* departmentId;

6 *protected* String gender;

7 *protected* String phone;

8

9 *public* *int* getStudentId() {

10 *return* studentId;

11 }

12 *public* *void* setStudentId(*int* studentId) {

13 *this*.studentId = studentId;

14 }

15 *public* String getName() {

16 *return* name;

17 }

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

19 *this*.name = name;

20 }

21 *public* *int* getDepartmentId() {

22 *return* departmentId;

23 }

24 *public* *void* setDepartmentId(*int* departmentId) {

25 *this*.departmentId = departmentId;

26 }

27 *public* String getGender() {

28 *return* gender;

29 }

30 *public* *void* setGender(String gender) {

31 *this*.gender = gender;

32 }

33 *public* String getPhone() {

34 *return* phone;

35 }

36 *public* *void* setPhone(String phone) {

37 *this*.phone = phone;

38 }

39

40

41 }

#### Hosteller.java

1 *public* *class* Hosteller *extends* Student

2 {

3 *private* String hostelName;

4 *private* *int* roomNumber;

5 *public* String getHostelName() {

6 *return* hostelName;

7 }

8 *public* *void* setHostelName(String hostelName) {

9 *this*.hostelName = hostelName;

10 }

11 *public* *int* getRoomNumber() {

12 *return* roomNumber;

13 }

14 *public* *void* setRoomNumber(*int* roomNumber) {

15 *this*.roomNumber = roomNumber;

16 }

17

18

19 }

#### Main.java

1 *import* java.util.\*;

2 *public* *class* Main

3 {

4 *public* *static* Hosteller getHostellerDetails()

5 {

6 Hosteller h1=*new* Hosteller();

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

8 System.out.println("Enter the Details:");

9 System.out.println("Student Id");

10 *int* sid=sc.nextInt();

11 System.out.println("Student Name");

12 String sname=sc.nextLine();

13 System.out.println("Department Id");

14 *int* did=sc.nextInt();

15 System.out.println("Gender");

16 String gen=sc.next();

17 System.out.println("Phone Number");

18 String ph=sc.next();

19 System.out.println("Hostel Name");

20 String hname=sc.nextLine();

21 System.out.println("Room Number");

22 *int* rno=sc.nextInt();

23 h1.setStudentId(sid);

24 h1.setName(sname);

25 h1.setDepartmentId(did);

26 h1.setGender(gen);

27 h1.setPhone(ph);

28 h1.setHostelName(hname);

29 h1.setRoomNumber(rno);

30 *return* h1;

31

32 }

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

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

35 Hosteller h=getHostellerDetails();

36 System.out.println("Modify Room Number(Y/N)");

37 *char* ans1=sc.next().charAt(0);

38 *if*(ans1=='Y' || ans1=='y')

39 {

40 System.out.println("New Room Number");

41 *int* nrno=sc.nextInt();

42 h.setRoomNumber(nrno);

43 }

44 System.out.println("Modify Phone Number(Y/N)");

45 *char* ans2=sc.next().charAt(0);

46 *if*(ans1=='Y' || ans1=='y')

47 {

48 System.out.println("New Phone Number");

49 String phone=sc.next();

50 h.setPhone(phone);

51 }

52 System.out.println("The Student Details");

53 System.out.println(h.getStudentId()+" "+h.getName()+" "+h.getDepartmentId()+" "+h.getGender()+" "+

54 h.getPhone()+" "+h.getHostelName()+" "+h.getRoomNumber());

55

56

57

58 }

59 }

## **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

1

2 *public* *class* Customer {

3

4

5 //Attributes

6 *private* *int* customerId;

7 *private* String customerName;

8 *private* String emailId;

9

10 //Constructor

11 *public* Customer(*int* customerId, String customerName, String emailId) {

12

13 *this*.customerId = customerId;

14 *this*.customerName = customerName;

15 *this*.emailId = emailId;

16 }

17

18 //Getters and Setters

19 *public* *int* getCustomerId() {

20 *return* customerId;

21 }

22

23 *public* *void* setCustomerId(*int* customerId) {

24 *this*.customerId = customerId;

25 }

26

27 *public* String getCustomerName() {

28 *return* customerName;

29 }

30

31 *public* *void* setCustomerName(String customerName) {

32 *this*.customerName = customerName;

33 }

34

35 *public* String getEmailId() {

36 *return* emailId;

37 }

38

39 *public* *void* setEmailId(String emailId) {

40 *this*.emailId = emailId;

41 }

42

43 }

44

#### Account.java

1 *public* *abstract* *class* Account {

2

3 *protected* *int* accountNumber;

4 *protected* Customer customerObj;

5 *protected* *double* balance;

6 *public* Account(*int* accno,Customer CustomerObj,*double* balance)

7

8 {

9 *this*.accountNumber=accno;

10 *this*.customerObj=CustomerObj;

11 *this*.balance=balance;

12 }

13

14 *public* *int* getAccountNumber() {

15 *return* accountNumber;

16 }

17

18 *public* *void* setAccountNumber(*int* accountNumber) {

19 *this*.accountNumber = accountNumber;

20 }

21

22 *public* Customer getCustomerObj() {

23 *return* customerObj;

24 }

25

26 *public* *void* setCustomerObj(Customer customerObj) {

27 *this*.customerObj = customerObj;

28 }

29

30 *public* *double* getBalance() {

31 *return* balance;

32 }

33

34 *public* *void* setBalance(*double* balance) {

35 *this*.balance = balance;

36 }

37 *abstract* *public* *boolean* withdraw(*double* amount);

38

39

40

41 }

#### SavingsAccount.java

1 *public* *class* SavingsAccount *extends* Account {

2 *public* SavingsAccount(*int* acntno,Customer cobj,*double* bal,*double* minbal)

3 { *super*(acntno,cobj,bal);

4 *this*.minimumBalance=minbal;

5 }

6

7

8 *private* *double* minimumBalance;

9 *public* *double* getMinimumBalance() {

10 *return* minimumBalance;

11 }

12

13 *public* *void* setMinimumBalance(*double* minimumBalance) {

14 *this*.minimumBalance = minimumBalance;

15 }

16 *public* *boolean* withdraw(*double* amount)

17 {

18 *if*(balance-amount>minimumBalance )

19 {balance=balance-amount;

20 *return* *true*;

21 }

22

23 *return* *false*;

24

25 }

26

27

28 }

29

30

#### Main.java

1 *import* java.util.\*;

2 *public* *class* Main{

3

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

5 Customer c=*new* Customer(123,"abc","jdhf");

6 c.setCustomerId(123);

7 c.setCustomerName("abc");

8 c.setEmailId("jegf");

9 System.out.println(c.getCustomerId());

10 System.out.println(c.getCustomerName());

11 System.out.println(c.getEmailId());

12

13

14 }

15 }