**Q1. Write a java program to implement “Student Management System”. In which you have to include at least three from the given topics as required: Inheritance, Overriding Methods, Polymorphism, Abstract Classes, Nested Classes, Interfaces, Lambda Expressions, Exceptional Handling and I/O Fundamentals.**

**Solution:**

import java.util.\*; //for scanner class

import java.io.\*; //for bufferedReader class

class Details //parent class

{

int rno;

String name;

Details(int r, String n) //Constructor of super class

{

rno=r;

name=n;

}

void putDetails() //Displaying basic details which will be overridden

{

System.out.println("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*The student details are:\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.println("\n\n\tRoll Number:"+rno);

System.out.println("\n\tName:"+name);

}

}

class PersonalInfo extends Details //Personal information inherits from details

{

String gender,phno,mail;

PersonalInfo(int r, String n, String g, String ph, String m) //Constructor of subclass Personal information

{

super(r,n); //calling the super class constructor

gender=g;

phno=ph;

mail=m;

}

void putDetails() //Displaying personal information by overriding the base class method

{

super.putDetails(); //Using parent class method to display basic details

System.out.println("\n\*\*\*\*\*\*\*\*\*The personal information of the student:\*\*\*\*\*\*\*\*\*\*");

System.out.println("\n\n\tGender:"+gender);

System.out.println("\n\tPhone Number:"+phno);

System.out.println("\n\tEmail Id:"+mail);

}

}

class AcademicDetails extends Details //Academic details inheriting from details

{

String dept, grade;

int sem;

float m1, m2, m3, avg;

AcademicDetails(int r, String n, String d,int s, float mrk1, float mrk2, float mrk3) //constructor

{

super(r,n); //calling parent class constructor

dept=d;

sem=s;

m1=mrk1;

m2=mrk2;

m3=mrk3;

}

void calc\_grade() //Calculating grade

{

avg=(m1+m2+m3)/3;

if(avg>=90)

grade="A";

else if(avg>=75)

grade="B";

else if(avg>=60)

grade="C";

else if(avg>=40)

grade="D";

else

grade="Fail";

}

void putDetails() //displaying academic details which overrides the base class method

{

super.putDetails();

System.out.println("\n\n\*\*\*\*\*\*\*\*\*The academic information of the student:\*\*\*\*\*\*\*\*\*\*");

System.out.println("\n\n\tDepartment:"+dept);

System.out.println("\n\tSemester:"+sem);

System.out.println("\n\tMarks in three Subjects are:\n\t\t\t\t\t"+m1+"\n\t\t\t\t\t"+m2+"\n\t\t\t\t\t"+m3);

System.out.println("\n\tAverage:"+avg);

System.out.println("\n\tGrade:"+grade);

}

}

class RollnoException extends Exception //User defined exception which raises if roll number is not found

{

int rno;

RollnoException(int r)

{

rno=r;

}

public String toString()

{

return("The entered roll number "+rno+" is not present");

}

}

class stud //class with main method

{

public static void main(String args[]) throws IOException

{

int i,n,choice,rollno;

char cont='y';

Scanner sc= new Scanner(System.in);

System.out.println("\n\n\t\t\t\*\*\*\*\*WELCOME TO STUDENT MANAGEMENT SYSTEM\*\*\*\*\*");

System.out.println("--------------------------------------------------------------------------------------------");

BufferedReader br= new BufferedReader(new InputStreamReader(System.in));

System.out.println("\n\*\*Enter the total number of students you want to have:\*\*");

n=sc.nextInt();

PersonalInfo pi[]= new PersonalInfo[n]; //creating an array of objects of type PersonalInfo

AcademicDetails ad[]= new AcademicDetails[n]; //creating an array of objects of type AcademicDetails

for(i=0;i<n;i++) //Taking the details of n students

{

System.out.println("\n\n\tLets have entries of the details of the Student "+(i+1));

System.out.println("\n# Enter the Roll Number: #");

int rno=sc.nextInt();

System.out.println("\n# Enter the Name: #");

String name=br.readLine();

System.out.println("\n# Enter the Gender (Male/Female): #");

String gender=br.readLine();

System.out.println("\n# Enter the Phone Number: #");

String phno=br.readLine();

System.out.println("\n# Enter the Email Id: #");

String mail=br.readLine();

System.out.println("\n# Enter the Department: #");

String dept=br.readLine();

System.out.println("\n# Enter the Semester: #");

int sem=sc.nextInt();

System.out.println("\n# Enter the marks obtained in THREE subjects: #");

float m1=sc.nextFloat();

float m2=sc.nextFloat();

float m3=sc.nextFloat();

pi[i]= new PersonalInfo(rno, name, gender, phno, mail);

ad[i]= new AcademicDetails(rno, name, gender, sem, m1, m2, m3);

}

while(cont=='y' || cont=='Y') //displaying required info

{

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

System.out.println("\n# 1.Print PERSONAL INFORMATION of a student. #\n \n# 2.Print ACADEMIC DETAILS of a student #\n\n# 3.Exit #");

choice=sc.nextInt();

if(choice==1 || choice==2)

{

System.out.println("\n\n# Enter the Roll number of the student /( PLEASE ENTER A VALID ONE /): # ");

rollno=sc.nextInt();

try

{

display(n,choice,rollno,pi,ad);

}

catch(RollnoException rne)

{

System.out.println("\n\n\t.....Exception Handled...."+rne);

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

}

}

else if(choice==2)

{

System.out.println("\n\n# Enter the Roll number of the student /( PLEASE ENTER A VALID ONE /): #");

rollno=sc.nextInt();

try

{

display(n,choice,rollno,pi,ad);

}

catch(RollnoException rne)

{

System.out.println("\n\n\t.....Exception Handled...."+rne);

}

}

else if(choice==3)

break;

else

System.out.println("\n\n# Please enter a VALID choice #");

System.out.println("\n\n\*\*Do you want to continue Y/N ??:\*\*");

cont=sc.next().charAt(0);

}

}

public static void display(int n,int choice, int rollno, PersonalInfo pi[], AcademicDetails ad[] ) throws RollnoException //Raising Exception

{

if(search\_rno(rollno,n,pi))

throw new RollnoException(rollno);

if(choice==1)

{

for(int i=0;i<n;i++)

{

if(pi[i].rno==rollno)

pi[i].putDetails();

}

}

else if(choice==2)

{

for(int i=0;i<n;i++)

{

if(ad[i].rno==rollno)

{

ad[i].calc\_grade();

ad[i].putDetails();

}

}

}

}

static boolean search\_rno(int r,int n, PersonalInfo pi[]) //Checks if the given roll number is present in the input array

{

int count=0;

for(int i=0;i<n;i++)

{

if(pi[i].rno==r)

{

count++;

break;

}

}

if(count==0)

return true;

else

return false;

}

}

**Console Screen:**

**A screenshot of a computer screen

Description automatically generated**

**A screenshot of a computer screen

Description automatically generated**

**A screenshot of a computer screen

Description automatically generated**

**A screenshot of a computer screen

Description automatically generated**

**A screenshot of a computer screen

Description automatically generated**

**Output Screens:**

**A screenshot of a cell phone

Description automatically generated**

**A screenshot of a cell phone

Description automatically generated**

**\*Entries of details**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**\*Personal Information.**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**\*Academic Informations.**

**A screenshot of a computer

Description automatically generated**

**\*Exception handled.**