Name: Krithika Swaminathan
Roll No.: 205001057

Assignment 3 - Inheritance

Q1: Create a class hierarchy for the classes defined below: Design a class called Person as described. A sub-class Student of class Person is designed. A sub-class Faculty of class Person is designed. Write a test driver called TestInheritance to test all the public methods that display the student and faculty details.
Use the following to calculate Net Salary:
Gross salary = Basicpay + DA as 60% of basic + HRA as 10% of basic
Deductions = Medical Insurance as 8.5% of basic + PF as 8% of basic
Net salary = Gross salary – Deductions
Use the following to calculate GPA

Grade	Point value for the grade
A	5
B	4
C	3
D	2
E	0

Grade point = credit * point value for the grade
GPA = Total points earned / Total credits

Code:

```
import java.util.Scanner;
class Person {
	//data members
	private int aadhaar;
	private String name;
	private String address;
	private char gender;

	//constructor
	public Person(int a, String n, String add, char g) {
		aadhaar = a;
		name = n;
		address = add;
		gender = g;
		}

	//methods
	public String getName() {
		return name;
		}

	public String getAddress() {
		return address;
```

```
		}

	public void setAddress(String add) {
		address = add;
		}

	public char getGender() {
		return gender;
		}

	void Display() {
		System.out.println("PERSON DETAILS:");
		System.out.println("Name: "+name);
		System.out.println("Aadhaar number: "+aadhaar);
		System.out.println("Address: "+address);
		System.out.println("Gender: "+gender);
		}
	}

class Student extends Person {
	//data members
	private String program;
	private int year;
	private char sub1_grade;
	private char sub2_grade;
	private char sub3_grade;
	private int sub1_credit;
	private int sub2_credit;
	private int sub3_credit;

	//constructor
	public Student(int a, String n, String add, char g, String p, int y, char s1g, char s2g, char s3g, int s1c,
int s2c, int s3c) {
		super(a,n,add,g);
		program = p;
		year = y;
		sub1_grade = s1g;
		sub2_grade = s2g;
		sub3_grade = s3g;
		sub1_credit = s1c;
		sub2_credit = s2c;
		sub3_credit = s3c;
		}

	//methods
	public String getProgram() {
		return program;
		}
```

```
public int getYear() {
        return year;
        }

public void setYear(int y) {
        year = y;
        }

public char getsub1_grade() {
        return sub1_grade;
        }

public char getsub2_grade() {
        return sub2_grade;
        }

public char getsub3_grade() {
        return sub3_grade;
        }

public int getsub1_credit() {
        return sub1_credit;
        }

public int getsub2_credit() {
        return sub2_credit;
        }

public int getsub3_credit() {
        return sub3_credit;
        }

public void setsub1_grade(char s1g) {
        sub1_grade = s1g;
        }

public void setsub2_grade(char s2g) {
        sub2_grade = s2g;
        }

public void setsub3_grade(char s3g) {
        sub3_grade = s3g;
        }

public void setsub1_credit(int s1c) {
        sub1_credit = s1c;
        }

public void setsub2_credit(int s2c) {
```

```
		sub2_credit = s2c;
		}

	public void setsub3_credit(int s3c) {
		sub3_credit = s3c;
		}

	public float calGPA() {
		int pointvalue[] = new int[] {5,4,3,2,0};
		int GPA = (pointvalue[(int)sub1_grade-65]*sub1_credit
			+ pointvalue[(int)sub2_grade-65]*sub2_credit
			+ pointvalue[(int)sub3_grade-65]*sub3_credit)
			/(sub1_credit+sub2_credit+sub3_credit);
		return GPA;
		}

	void Display() {
		System.out.println("GRADE SHEET:");
		System.out.println("Name: "+getName());
		System.out.println("Address: "+getAddress());
		System.out.println("Program: "+program);
		System.out.println("Year: "+year);
		System.out.println("GPA: "+calGPA());
		}
	}

class Faculty extends Person {
	private String designation;
	private String department;
	private float basicpay;

	public Faculty(int a, String n, String add, char g, String desig, String dept, float pay) {
		super(a, n, add, g);
		designation = desig;
		department = dept;
		basicpay = pay;
		}

	public String getDesig() {
		return designation;
		}

	public void setDesig(String desig) {
		designation = desig;
		}

	public float getBasic() {
		return basicpay;
		}
```

```
	public void setBasic(float bp) {
		basicpay = bp;
		}

	public double calSalary() {
		double DA = 0.6*basicpay, HRA = 0.1*basicpay, MedIns = 0.085*basicpay, PF =
0.08*basicpay;
		double GS = basicpay + DA + HRA;
		double Deduction = MedIns + PF;
		return GS - Deduction;
		}

	void Display() {
		System.out.println("FACULTY PAY SLIP:");
		System.out.println("Name: "+getName());
		System.out.println("Address: "+getAddress());
		System.out.println("Department: "+department);
		System.out.println("Designation: "+designation);
		System.out.println("Salary: "+calSalary());
		}
	}

class TestInheritance {
	public static void main (String a[]) {
		//declaring new scanner object
		Scanner sc = new Scanner(System.in);

		//getting data from user
		int aadhaar;
		String address, name;
		char gender;
		System.out.println("__PERSON__");
		System.out.print("Enter aadhaar number: ");
		aadhaar = sc.nextInt();
		sc.nextLine();
		System.out.print("Enter name: ");
		name = sc.nextLine();
		System.out.print("Enter address: ");
		address = sc.nextLine();
		System.out.print("Enter gender: ");
		gender = sc.next().charAt(0);

		Person P = new Person(aadhaar, name, address, gender);
		//display
		P.Display();

		//Student
		System.out.println("\n__STUDENT__");
```

```
String program;
int year, s1c, s2c, s3c;
char s1g, s2g, s3g;
System.out.print("Enter aadhaar number: ");
aadhaar = sc.nextInt();
sc.nextLine();
System.out.print("Enter name: ");
name = sc.nextLine();
System.out.print("Enter address: ");
address = sc.nextLine();
System.out.print("Enter gender: ");
gender = sc.next().charAt(0);
sc.nextLine();
System.out.print("Enter program: ");
program = sc.nextLine();
System.out.print("Enter year: ");
year = sc.nextInt();
System.out.print("Enter subject 1 credit: ");
s1c = sc.nextInt();
System.out.print("Enter subject 2 credit: ");
s2c = sc.nextInt();
System.out.print("Enter subject 3 credit: ");
s3c = sc.nextInt();
System.out.print("Enter subject 1 grade: ");
s1g = sc.next().charAt(0);
System.out.print("Enter subject 2 grade: ");
s2g = sc.next().charAt(0);
System.out.print("Enter subject 3 grade: ");
s3g = sc.next().charAt(0);

Student S = new Student(aadhaar, name, address, gender, program, year, s1g, s2g, s3g, s1c,
s2c, s3c);
//display
S.Display();

//Faculty
System.out.println("\n__FACULTY__");
String Dept,Desig;
float bp;
System.out.print("Enter aadhaar number: ");
aadhaar = sc.nextInt();
sc.nextLine();
System.out.print("Enter name: ");
name = sc.nextLine();
System.out.print("Enter address: ");
address = sc.nextLine();
System.out.print("Enter gender: ");
gender = sc.next().charAt(0);
sc.nextLine();
```

```
            System.out.print("Enter Department: ");
            Dept = sc.nextLine();
            System.out.print("Enter Designation: ");
            Desig = sc.nextLine();
            System.out.print("Enter Basic Pay: ");
            bp = sc.nextFloat();

            Faculty F = new Faculty(aadhaar, name, address, gender, Desig, Dept, bp);
            //display
            F.Display();
            }
    }
```

Output:

```
kri@kri-ubuntu:~/workspace$ javac TestInheritance.java
kri@kri-ubuntu:~/workspace$ java TestInheritance
__PERSON__
Enter aadhaar number: 3828
Enter name: Surya Ganesh
Enter address: 42, Besant Avenue, Chennai
Enter gender: M
PERSON DETAILS:
Name: Surya Ganesh
Aadhaar number: 3828
Address: 42, Besant Avenue, Chennai
Gender: M

__STUDENT__
Enter aadhaar number: 9389
Enter name: Aruna Chaudhary
Enter address: 31, Brown Brick Rd., New Delhi
Enter gender: F
Enter program: ECE
Enter year: 3
Enter subject 1 credit: 4
Enter subject 2 credit: 3
Enter subject 3 credit: 3
Enter subject 1 grade: A
Enter subject 2 grade: C
Enter subject 3 grade: B
GRADE SHEET:
Name: Aruna Chaudhary
Address: 31, Brown Brick Rd., New Delhi
Program: ECE
Year: 3
GPA: 4.0
```

Name: Krithika Swaminathan
Roll No.: 205001057

```
__FACULTY__
Enter aadhaar number: 8271
Enter name: Indra Kumar
Enter address: 10, Chinna Salai, Kumbakonam
Enter gender: F
Enter Department: Chemical
Enter Designation: Assistant Professor
Enter Basic Pay: 32000
FACULTY PAY SLIP:
Name: Indra Kumar
Address: 10, Chinna Salai, Kumbakonam
Department: Chemical
Designation: Assistant Professor
Salary: 49120.0
```

Name: Krithika Swaminathan
Roll No.: 205001057

Q2: Create a class hierarchy for the classes as defined below: Design a class Shape as described. A sub-class Circle of class Shape is designed as shown. A sub-class Rectangle of class Shape is designed as shown. A sub-class Square of class Rectangle is designed as shown. Write a test driver called TestShape to test all the public methods. Use an array of objects of type Shape and display the area and perimeter of all the shapes (Circle, Rectangle and Square).
Note down the scope of the variable declared as protected.

Code:

```
import java.util.Scanner;

class Shape {
	//data members
	protected String color;
	//constructors
	Shape() { color = "red"; }
	Shape(String col) { color = col; }
	//public methods
	String getColor() { return color; }
	void setColor(String col) { color = col; }

	//defining dummy methods to facilitate method overriding
	float getRadius() { return 0; }
	void setRadius(float none) {}
	float getWidth() {return 0; }
	void setWidth(float none) {}
	float getLength() { return 0; }
	void setLength(float none) {}
	float getSide() { return 0; }
	void setSide(float none) {}
	double getArea() { return 0; }
	double getPerimeter() { return 0; }
	}

class Circle extends Shape {
	//data members
	protected float radius;
	//constructors
	Circle() {
		super();
		radius = 1;
		}
	Circle(float r) {
		super();
		radius = r;
		}
```

```
	Circle(float r, String col) {
		super(col);
		radius = r;
		}
	//public methods
	float getRadius() { return radius; }
	void setRadius(float r) { radius = r; }
	double getArea() {
		return 3.14*radius*radius;
		}
	double getPerimeter() {
		return 2*3.14*radius;
		}
	}

class Rectangle extends Shape {
	//data members
	protected float width;
	protected float length;
	//constructors
	Rectangle() {
		super();
		width = 1;
		length = 1;
		}
	Rectangle(float w, float l) {
		super();
		width = w;
		length = l;
		}
	Rectangle(float w, float l, String col) {
		super(col);
		width = w;
		length = l;
		}
	//public methods
	float getWidth() { return width; }
	void setWidth(float w) { width = w; }
	float getLength() { return length; }
	void setLength(float l) { length = l; }
	double getArea() {
		return length*width;
		}
	double getPerimeter() {
		return 2*(length+width);
		}
	}

class Square extends Rectangle {
```

```
	//constructors
	Square() {
		super();
		}
	Square(float side) {
		super(side,side);
		}
	Square(float side, String col) {
		super(side,side,col);
		}
	//public methods
	float getSide() { return length; }
	void setSide(float side) {
		length = side;
		width = side;
		}
	}

class TestShape {
	public static void main (String a[]) {
		//declaring new scanner object
		Scanner sc = new Scanner(System.in);

		//declaring a 2D array of shapes - each row contains a different shape
		Shape shapes[][] = new Shape[3][3];

		shapes[0][0] = new Circle();
		shapes[0][1] = new Circle(2);
		shapes[0][2] = new Circle(3,"blue");

		shapes[1][0] = new Rectangle();
		shapes[1][1] = new Rectangle(5,7);
		shapes[1][2] = new Rectangle(8,3,"yellow");

		shapes[2][0] = new Square();
		shapes[2][1] = new Square(4);
		shapes[2][2] = new Square(2,"purple");

		//displaying values
		for (int i=0; i<shapes.length; i++) {
			if (i==0) System.out.println("CIRCLES\n");
			else if (i==1) System.out.println("RECTANGLES\n");
			else System.out.println("SQUARES\n");

			for (int j=0; j<shapes.length; j++) {
				System.out.println("Colour: "+shapes[i][j].getColor());
				switch(i) {
					case 0: {
						System.out.println("Radius: "+shapes[i][j].getRadius());
```

```
                        break;
                        }
                    case 1: {
                        System.out.println("Width: "+shapes[i][j].getWidth());
                        System.out.println("Length: "+shapes[i][j].getLength());
                        break;
                        }
                    case 2: {
                        System.out.println("Side: "+shapes[i][j].getSide());
                        break;
                        }
                    default: System.exit(0);
                    }
                System.out.println("Area: "+shapes[i][j].getArea());
                System.out.println("Perimeter: "+shapes[i][j].getPerimeter()+'\n');
                }
        }

    //getting new values from the user
    System.out.println("\nEnter new values: ");
    System.out.print("Enter new color: ");
    String col = sc.next();
    System.out.print("Enter new radius: ");
    float rad = sc.nextFloat();
    System.out.print("Enter new width: ");
    float wid = sc.nextFloat();
    System.out.print("Enter new length: ");
    float len = sc.nextFloat();

    //setting new values for objects
    System.out.println("\nNew values:");
    for (int i=0; i<shapes.length; i++) {
        if (i==0) System.out.println("CIRCLES\n");
        else if (i==1) System.out.println("RECTANGLES\n");
        else System.out.println("SQUARES\n");

        for (int j=0; j<shapes.length; j++) {
            shapes[i][j].setColor(col);
            System.out.println("Colour: "+shapes[i][j].getColor());
            switch(i) {
                case 0: {
                    shapes[i][j].setRadius(rad);
                    System.out.println("Radius: "+shapes[i][j].getRadius());
                    break;
                    }
                case 1: {
                    shapes[i][j].setWidth(wid);
                    shapes[i][j].setLength(len);
                    System.out.println("Width: "+shapes[i][j].getWidth());
```

```
                                        System.out.println("Length: "+shapes[i][j].getLength());
                                        break;
                                        }
                                case 2: {
                                        shapes[i][j].setSide(len);
                                        System.out.println("Side: "+shapes[i][j].getSide());
                                        break;
                                        }
                                default: System.exit(0);
                                }
                        System.out.println("Area: "+shapes[i][j].getArea());
                        System.out.println("Perimeter: "+shapes[i][j].getPerimeter()+'\n');
                        }
                }
        }
}
```

Output:

```
kri@kri-ubuntu:~/workspace$ javac TestShape.java
kri@kri-ubuntu:~/workspace$ java TestShape
CIRCLES

Colour: red
Radius: 1.0
Area: 3.14
Perimeter: 6.28

Colour: red
Radius: 2.0
Area: 12.56
Perimeter: 12.56

Colour: blue
Radius: 3.0
Area: 28.259999999999998
Perimeter: 18.84

RECTANGLES

Colour: red
Width: 1.0
Length: 1.0
Area: 1.0
Perimeter: 4.0

Colour: red
Width: 5.0
Length: 7.0
Area: 35.0
Perimeter: 24.0

Colour: yellow
Width: 8.0
Length: 3.0
Area: 24.0
Perimeter: 22.0
```

Name: Krithika Swaminathan
Roll No.: 205001057

```
SQUARES

Colour: red
Side: 1.0
Area: 1.0
Perimeter: 4.0

Colour: red
Side: 4.0
Area: 16.0
Perimeter: 16.0

Colour: purple
Side: 2.0
Area: 4.0
Perimeter: 8.0


Enter new values:
Enter new color: green
Enter new radius: 5
Enter new width: 2
Enter new length: 2
```

```
New values:
CIRCLES

Colour: green
Radius: 5.0
Area: 78.5
Perimeter: 31.400000000000002

Colour: green
Radius: 5.0
Area: 78.5
Perimeter: 31.400000000000002

Colour: green
Radius: 5.0
Area: 78.5
Perimeter: 31.400000000000002

RECTANGLES

Colour: green
Width: 2.0
Length: 2.0
Area: 4.0
Perimeter: 8.0

Colour: green
Width: 2.0
Length: 2.0
Area: 4.0
Perimeter: 8.0

Colour: green
Width: 2.0
Length: 2.0
Area: 4.0
Perimeter: 8.0
```

Name: Krithika Swaminathan
Roll No.: 205001057

- The scope of the variables declared as protected is the class in which it is defined and the derived classes of the same class.