Create a class hierarchy for the classes defined below:

Design a class called Person as described below:

Person

-aadhaar:int

-name:String

-address:String-gender:char

+Person(aadhaar,name,address,gender)

+getName():String

+getAddress():String

+setAddress(address):void

+getGender():char

A sub-class Student of class Person is designed as shown below:

Student

-program:String

-year:int

-totalmark:float

+Student(aadhaar,name,address,gender,program

,year,total)

+getProgram():String

+getYear():int

+setYear(year):void

+getTotal():float

+setTotal(tot):void

+calGPA():float

A sub-class Faculty of class Person is designed as shown below:

Faculty

-designation:String

-department:String

-basicpay:float

+Faculty(aadhaar,name,address,gender,designati

on,dept,pay)

+getDesig():String

+setDesig(desig):void

+setBasic(basic):void

+getBasic():float

+calSalary():float

Note the following:

1. The hierarchy Person -> Student or Person -> Faculty is a Single-level inheritance

type.

2. The type of above entire class hierarchy is the Hierarchical Inheritance.3. Note the use of constructors at all levels of class hierarchy.

EXERCISE : 3A

1. Draw the class diagram of the above class hierarchy.

2. 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

import java.util.Scanner;

class Person{

private int aadhaar;

private String name;

private String address;

private char gender;

//Constructor

public Person(int aadhaar,String name,String address,char gender){

this.aadhaar=aadhaar;

this.name=name;

this.address=address;

this.gender=gender;

}

//Return Aadhaar

public int getAadhaar(){

return aadhaar;

}

//Return Name

public String getName(){

return name;

}

//Return Address

public String getAddress(){

return address;

}

//Set Address

public void setAddress(String address){

this.address=address;

}

//Return Gender

public char getGender(){

return gender;

}

}

class Student extends Person{

private String program;

private int year;

private float totalmark;

public Student(int aadhaar,String name,String address,char gender,String program,int year,float total){

super(aadhaar,name,address,gender);

this.program=program;

this.year=year;

this.totalmark=total;

}

//Return Program

public String getProgram(){

return program;

}

//Return year

public int getYear(){

return year;

}

//Set Year

public void setYear(int year){

this.year=year;

}

//Return Total

public float getTotal(){

return totalmark;

}

//Set Total

public void setTotal(float tot){

this.totalmark=tot;

}

//Return GPA

public float calGPA(){

return totalmark/10;

}

}

class Faculty extends Person{

private String designation;

private String department;

private double basicpay;

public Faculty(int aadhaar,String name,String address,char gender,String designation,String department,double basicpay){

super(aadhaar,name,address,gender);

this.designation=designation;

this.department=department;

this.basicpay=basicpay;

}

//Return Designation

public String getDesig(){

return designation;

}

//Return Department

public String getDept(){

return department;

}

//Set Designation

public void setDesig(String desig){

designation=desig;

}

//Set Basic Pay

public void setBasic(double basic){

basicpay=basic;

}

//Return Basic Pay

public double getBasic(){

return basicpay;

}

//Return Salary

public double calSalary(){

double GP,Ded,NP;

GP=basicpay+(0.6\*basicpay)+(0.1\*basicpay);

Ded=(0.085\*basicpay)+(0.08\*basicpay);

NP=GP-Ded;

return NP;

}

}

public class TestInheritance{

public static void main(String args[]){

int aadhaar;

String name=new String();

String address=new String();

char gender;

String program=new String();

int year;

float totalmark;

String designation=new String();

String department=new String();

double basicpay;

Scanner in=new Scanner(System.in);

System.out.println("Enter Student Details: ");

System.out.print("Enter name: ");name=in.nextLine();

System.out.print("Enter aadhaar number: ");aadhaar=in.nextInt();

in.nextLine();

System.out.print("Enter address: ");address=in.nextLine();

System.out.print("Enter gender: ");gender=in.next().charAt(0);

in.nextLine();

System.out.print("Enter program: ");program=in.nextLine();

System.out.print("Enter year: ");year=in.nextInt();

System.out.print("Enter total marks: ");totalmark=in.nextFloat();

in.nextLine();

Student s=new Student(aadhaar,name,address,gender,program,year,totalmark);

s.setYear(year);s.setAddress(address);s.setTotal(totalmark);

System.out.println("Enter Faculty Details: ");

System.out.print("Enter name: ");name=in.nextLine();

System.out.print("Enter aadhaar number: ");aadhaar=in.nextInt();

in.nextLine();

System.out.print("Enter address: ");address=in.nextLine();

System.out.print("Enter gender: ");gender=in.next().charAt(0);

in.nextLine();

System.out.print("Enter designation: ");designation=in.nextLine();

System.out.print("Enter department: ");department=in.nextLine();

System.out.print("Enter basic pay: ");basicpay=in.nextDouble();

Faculty f=new Faculty(aadhaar,name,address,gender,designation,department,basicpay);

f.setDesig(designation);f.setBasic(basicpay);

System.out.println("Student Details: ");

System.out.println("Name: "+s.getName());

System.out.println("Aadhar: "+s.getAadhaar());

System.out.println("Address: "+s.getAddress());

System.out.println("Gender: "+s.getGender());

System.out.println("Program: "+s.getProgram());

System.out.println("Year: "+s.getYear());

System.out.println("Total Marks: "+s.getTotal());

System.out.println("GPA: "+s.calGPA());

System.out.println();

System.out.println("Faculty Details: ");

System.out.println("Name: "+f.getName());

System.out.println("Aadhar: "+f.getAadhaar());

System.out.println("Address: "+f.getAddress());

System.out.println("Gender: "+f.getGender());

System.out.println("Designation: "+f.getDesig());

System.out.println("Department: "+f.getDept());

System.out.println("Salary: "+f.calSalary());

}

}

/\*

Output:

Enter Student Details:

Enter name: A

Enter aadhaar number: 1234

Enter address: asdf

Enter gender: m

Enter program: sdfg

Enter year: 2

Enter total marks: 98.5

Enter Faculty Details:

Enter name: B

Enter aadhaar number: 9876

Enter address: lkjh

Enter gender: f

Enter designation: kjhg

Enter department: mnbv

Enter basic pay: 2000.0

Student Details:

Name: A

Aadhar: 1234

Address: asdf

Gender: m

Program: sdfg

Year: 2

Total Marks: 98.5

GPA: 9.85

Faculty Details:

Name: B

Aadhar: 9876

Address: lkjh

Gender: f

Designation: kjhg

Department: mnbv

Salary: 3070.0

\*/

Create a class hierarchy for the classes as defined below:

Design a class Shape as described below:

Shape

#color:String=”red”

+Shape()

+Shape(color)

+getColor():String

+setColor(color):void

A sub-class Circle of class Shape is designed as shown below:

Circle

#radius:float=1.0

+Circle()

+Circle(radius)

+Circle(radius,color)

+getRadius():float

+setRadius(radius):void

+getArea():float

# - protected+getPerimeter():float

A sub-class Rectangle of class Shape is designed as shown below:

Rectangle

#width:float=1.0

#length:float=1.0

+Rectangle()

+Rectangle(width,length)

+Rectangle(width,length,color)

+getWidth():float

+setWidth(width):void

+getLength():float

+setLength(length):void

+getArea():float

+getPerimeter():float

A sub-class Square of class Rectangle is designed as shown below:

Square

+Square()

+Square(side)

+Square(side,color)

+getSide():float

+setSide(side):void

Note the following:

1. The hierarchy Shape --> Rectangle --> Square is a Multi-level inheritance type.

2. The type of above entire class hierarchy is the Hierarchical inheritance.

3. Note the constructor overloading at all the levels.

4. # denotes protected variable. The protected variables can be accessed by its

subclasses and classes in the same package.

EXERCISE : 3B

1. Draw the class diagram of the above class hierarchy.2. Write a test driver called TestShape to test all the public methods. Display the

area and perimeter of all the shapes (Circle, Rectangle and Square).

3. Note down the scope of the variable declared as protected.

import java.util.Scanner;

class Shape{

protected String color;

public Shape(){

color="red";

}

public Shape(String color){

this.color=color;

}

public String getColor(){

return color;

}

public void setColor(String color){

this.color=color;

}

}

class Circle extends Shape{

protected double radius;

public Circle(){

radius=1.0;

}

public Circle(double radius){

this.radius=radius;

}

public Circle(double radius, String color){

super(color);

this.radius=radius;

}

public double getRadius(){

return radius;

}

public void setRadius(double radius){

this.radius=radius;

}

public double getArea(){

return (double)3.14\*radius\*radius;

}

public double getPerimeter(){

return (double)2\*3.14\*radius;

}

}

class Rectangle extends Shape{

protected double width;

protected double length;

public Rectangle(){

width=1.0;

length=1.0;

}

public Rectangle(double width,double length){

this.width=width;

this.length=length;

}

public Rectangle(double width,double length,String color){

super(color);

this.width=width;

this.length=length;

}

public Rectangle(String color){

super(color);

}

public double getWidth(){

return width;

}

public void setWidth(double width){

this.width=width;

}

public double getLength(){

return length;

}

public void setLength(){

this.length=length;

}

public double getArea(){

return (double)width\*length;

}

public double getPerimeter(){

return 2\*(width+length);

}

}

class Square extends Rectangle{

protected double side;

public Square(){

side=1.0;

}

public Square(double side){

this.side=side;

}

public Square(double side,String color){

super(color);

this.side=side;

}

public double getSide(){

return side;

}

public void setSide(double side){

this.side=side;

}

}

public class TestShape{

public static void main(String args[]){

String color;

double radius;

double width;

double length;

double side;

Scanner in=new Scanner(System.in);

System.out.print("Enter color of Circle: ");

color=in.nextLine();

System.out.print("Enter radius of Circle: ");

radius=in.nextDouble();

in.nextLine();

Circle c=new Circle(radius,color);

c.setRadius(radius);c.setColor(color);

System.out.print("Enter color of Rectangle: ");

color=in.nextLine();

System.out.print("Enter width of Rectangle: ");

width=in.nextDouble();

System.out.print("Enter length of Rectangle: ");

length=in.nextDouble();

in.nextLine();

Rectangle r=new Rectangle(width,length,color);

System.out.print("Enter color of Square: ");

color=in.nextLine();

System.out.print("Enter side of Square: ");

side=in.nextDouble();

Square s=new Square(side,color);

System.out.println();

System.out.println("Colour of Circle:"+c.getColor());

System.out.println("Area of Circle :"+c.getArea());

System.out.println("Perimeter of Circle :"+c.getPerimeter());

System.out.println();

System.out.println("Colour of Rectangle:"+r.getColor());

System.out.println("Length of Rectangle :"+r.getLength());

System.out.println("Width of Rectangle :"+r.getWidth());

System.out.println("Area of Rectangle :"+r.getArea());

System.out.println("Perimeter of Rectangle :"+r.getPerimeter());

System.out.println();

System.out.println("Colour of Square:"+s.getColor());

System.out.println("Area of Square :"+s.getSide()\*s.getSide());

System.out.println("Perimeter of Square :"+4\*s.getSide());

}

}

/\*

Output:

Enter color of Circle: red

Enter radius of Circle: 5

Enter color of Rectangle: blue

Enter width of Rectangle: 6

Enter length of Rectangle: 4

Enter color of Square: green

Enter side of Square: 7

Color of Circle :red

Area of Circle :78.5

Perimeter of Circle :31.400000000000002

Color of Rectangle :blue

Length of Rectangle :6.0

Width of Rectangle :4.0

Area of Rectangle :24.0

Perimeter of Rectangle :20.0

Color of Square :green

Area of Square :49.0

Perimeter of Square :28.0

\*/