1.test inheritance using interface

import java.util.Scanner;

class Person

{

private String name,address;

Person(String b,String c)

{

name=b;

address=c;

}

public String getName()

{

return name;

}

public String getAddress()

{

return address;

}

public void setAddress(String addr)

{

address=addr;

}

}

interface Student

{

public double[] getMarks();

public double calGPA();

}

class Employee extends Person

{

private String empid,dept;

private double bp;

Employee(String b,String c,String e,String f,double g)

{

super(b,c);

empid=e;

dept=f;

bp=g;

}

public String getempid()

{

return empid;

}

public void setdept(String d)

{

dept=d;

}

public String getdept()

{

return dept;

}

public void setbasic(double d)

{

bp=d;

}

public double getbasic()

{

return bp;

}

public double calsalary()

{

double gross,deduc,net;

gross=bp\*1.7;

deduc=bp\*0.165;

net=gross-deduc;

return net;

}

}

class ResearchAssistant extends Employee implements Student

{

private String project,course;

private double totalmark,marks[]= new double[5];

ResearchAssistant(String b,String c,String e,String f,double g,String h,String i)

{

super(b,c,e,f,g);

project=h;

course=i;

totalmark=0;

marks=getMarks();

}

public String getproject()

{

return project;

}

public String getcourse()

{

return course;

}

public void setcourse(String c)

{

course=c;

}

public double gettotal()

{

return totalmark;

}

public void settotal()

{

double d=0;

for(int i=0;i<5;i++)

d+=marks[i];

totalmark=d;

}

public double calGPA()

{

return totalmark/50;

}

public double[] getMarks()

{

Scanner obj =new Scanner(System.in);

double a[]=new double[5];

for(int i=0;i<5;i++)

{

a[i]=obj.nextDouble();

}

return a;

}

}

class Faculty extends Employee

{

private String desig,course;

Faculty(String b,String c,String e,String f,double g,String h,String i)

{

super(b,c,e,f,g);

desig=h;

course=i;

}

public String getdesig()

{

return desig;

}

public void setdesig(String d)

{

desig=d;

}

public void setcourse(String d)

{

course=d;

}

public String getcourse()

{

return course;

}

}

public class Testpolyempl

{

public static void main(String Args[])

{

String a;

String b;

String c;

String d;

String e,t;

double g;

Scanner obj=new Scanner(System.in);

System.out.println("Enter Research Assistant details");

System.out.print("Name : ");

a=obj.next();

System.out.print("Address : ");

b=obj.next();

System.out.print("Emp ID : ");

c=obj.next();

System.out.print("Department : ");

d=obj.next();

System.out.print("basic : ");

g=obj.nextDouble();

System.out.print("Project : ");

e=obj.next();

System.out.print("Course: ");

t=obj.next();

ResearchAssistant s=new ResearchAssistant(a,b,c,d,g,e,t);

System.out.println("Enter Faculty details");

System.out.print("Name : ");

a=obj.next();

System.out.print("Address : ");

b=obj.next();

System.out.print("Emp ID : ");

c=obj.next();

System.out.print("Department : ");

d=obj.next();

System.out.print("basic : ");

g=obj.nextDouble();

System.out.print("Designation : ");

e=obj.next();

System.out.print("Course: ");

t=obj.next();

Faculty F=new Faculty(a,b,c,d,g,e,t);

System.out.println(" TRYING ALL FUNCTIONS");

System.out.println("1.ALL SET FUNCTION OF RESEARCH ASSISTANT");

s.settotal();

System.out.println("Enter course : ");

e=obj.next();

s.setcourse(e);

System.out.println("Enter basic : ");

g=obj.nextDouble();

s.setbasic(g);

System.out.println("2.ALL GET FUNCTIONS OF Student");

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

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

System.out.println("Emp ID : "+s.getempid());

System.out.println("Department : "+s.getdept());

System.out.println("Project : "+s.getproject());

System.out.println("Course : "+s.getcourse());

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

System.out.println("Salary : "+s.calsalary());

System.out.println("3.gpa calculation ");

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

System.out.println("1.ALL SET FUNCTION OF FACULTY");

System.out.println("enter basic : ");

g=obj.nextDouble();

F.setbasic(g);

System.out.println("enter designation : ");

b=obj.next();

F.setdesig(b);

System.out.println("Enter course : ");

e=obj.next();

s.setcourse(e);

System.out.println("2.ALL GET FUNCTIONS OF FACULTY");

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

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

System.out.println("Emp ID : "+s.getempid());

System.out.println("Department : "+s.getdept());

System.out.println("Designation : "+F.getdesig());

System.out.println("Basic Pay : "+F.getbasic());

System.out.println("Course : "+s.getcourse());

System.out.println("3.salary calculation ");

System.out.println("Net salary : "+F.calsalary());

}

}

2. Test Shape using Abstract Class

import java.util.\*;

abstract class Shape

{

protected String colour="RED";

Shape()

{

colour="BLUE";

}

Shape(String c)

{

colour=c;

}

String getColour()

{

return colour;

}

void setColour(String c)

{

colour=c;

}

abstract double getArea();

abstract double getPerimeter();

}

class Circle extends Shape

{

protected double radius=1.0;

Circle()

{

radius =1.0;

}

Circle(double r)

{

radius=r;

}

Circle(double r,String c)

{

super(c);

radius=r;

}

double getRadius()

{

return radius;

}

void setRadius(double r)

{

radius=r;

}

double getArea()

{

return (3.14\*radius\*radius);

}

double getPerimeter()

{

return (6.28\*radius);

}

}

class Rectangle extends Shape

{

protected double width=1.0,length=1.0;

Rectangle()

{

length =1.0;width=2.0;

}

Rectangle(double l,double w)

{

length=l;

width=w;

}

Rectangle(double l,double w,String c)

{

super(c);

length=l;

width=w;

}

double getWidth()

{

return width;

}

double getLength()

{

return length;

}

void setWidth(double r)

{

width=r;

}

void setLength(double r)

{

length=r;

}

double getArea()

{

return (length\*width);

}

double getPerimeter()

{

return (2\*(length+width));

}

}

class Square extends Rectangle

{

Square()

{

super(1.0,1.0);

}

Square(double s)

{

super(s,s);

}

Square(double s,String c)

{

super(s,s,c);

}

double getSide()

{

return getLength();

}

void setSide(double s)

{

setLength(s);

setWidth(s);

}

}

public class TestShape

{

public static void main(String args[])

{

Shape S[]=new Shape[80];

Scanner s=new Scanner(System.in);

int ch,ch1,c=0;

double x,y;

String cl;

do

{

System.out.print("\nMenu:\n1.Circle\n2.Rectangle\n3.Square\n4.Exit\n\nChoice: ");

ch=s.nextInt();

switch(ch)

{

case 1: System.out.print("\nCircle: 1.Default 2.Change radius 3. Change Radius and Colour \nChoice: ");ch1=s.nextInt();

if(ch1==1)

{

S[c]=new Circle();

}

else if(ch1==2) CVWQFE wwwQ

{

System.out.print("Radius: ");x=s.nextDouble();

S[c]=new Circle(x);

}

else

{

s.nextLine();System.out.print("Colour: ");cl=s.nextLine();

System.out.print("Radius: ");x=s.nextDouble();

S[c]=new Circle(x,cl);

}

System.out.println("Area of the circle is: "+S[c].getArea()+"\nPerimeter of the circle is: "+S[c].getPerimeter());

break;

case 2:

System.out.print("\nRectangle: 1.Default 2.Change dimensions 3. Change dimensions and Colour \nChoice: ");ch1=s.nextInt();

if(ch1==1)

{

S[c]=new Rectangle();

}

else if(ch1==2)

{

System.out.print("Length: ");x=s.nextDouble();

System.out.print("Width: ");y=s.nextDouble();

S[c]=new Rectangle(x,y);

}

else

{

s.nextLine();System.out.print("Colour: ");cl=s.nextLine();

System.out.print("Length: ");x=s.nextDouble();

System.out.print("Width: ");y=s.nextDouble();

S[c]=new Rectangle(x,y,cl);

}

System.out.println("Area of the Rectangle is: "+S[c].getArea()+"\nPerimeter of the Rectangle is: "+S[c].getPerimeter());

break;

case 3:

System.out.print("\nCircle: 1.Default 2.Change side 3. Change side and Colour \nChoice: ");ch1=s.nextInt();

if(ch1==1)

{

S[c]=new Square();

}

else if(ch1==2)

{

System.out.print("Side: ");x=s.nextDouble();

S[c]=new Square(x);

}

else

{

s.nextLine();System.out.print("Colour: ");cl=s.nextLine();

System.out.print("Side: ");x=s.nextDouble();

S[c]=new Square(x,cl);

}

System.out.println("Area of the Square is: "+S[c].getArea()+"\nPerimeter of the Square is: "+S[c].getPerimeter());

break;

}

}while(ch!=4);

}

}