**Java Assignment 6 : Part - 1**

Inheritance: Dynamic Method Dispatch Example in Part 1

By:

Vaibhav Jaiswal

PRN:17070122071

Batch:CS-4

Branch: CSE

SY CSE (Batch of 2017-2021)

**CODE**

import java.lang.\* ;

import java.util.\*;

abstract class Shape{

protected int dim1 , dim2 , dim3 , numSides ;

Shape(){

this(0,0,0,0) ;

}

Shape(int dim1 , int dim2 , int dim3 , int numSides){

this.dim1 = dim1 ;

this.dim2 = dim2 ;

this.dim3 = dim3 ;

this.numSides = numSides ;

}

public int getSides(){

return this.numSides ;

}

abstract public double getArea() ;

abstract public double getPerimeter() ;

}

class Rectangle extends Shape{

Rectangle(){

super() ;

}

Rectangle(int dim1 , int dim2){

super(dim1 , dim2 , 0 , 4) ;

}

public double getArea(){

return (super.dim1 \* super.dim2) ;

}

public double getPerimeter(){

return (2\*(super.dim1 + super.dim2)) ;

}

}

class RtTriangle extends Shape{

RtTriangle(){

super(0,0,0,0) ;

}

RtTriangle(int dim1 , int dim2){

super(dim1 , dim2 , 0 , 3) ;

}

public double getArea(){

return (0.5 \* super.dim1 \* super.dim2) ;

}

public double getPerimeter(){

return (super.dim1 + super.dim2 + Math.sqrt(Math.pow(dim1, 2)+ Math.pow(super.dim2, 2))) ;

}

}

class Circle extends Shape{

Circle(){

super(0,0,0,0) ;

}

Circle(int dim1){

super(dim1 , 0 , 0 , 0) ;

}

public double getArea(){

return (Math.PI \* Math.pow(super.dim1, 2)) ;

}

public double getPerimeter(){

return (2\*Math.PI\*super.dim1) ;

}

}

class Driver{

public static void main(String[] args) {

Shape shape ;

shape = new Rectangle(5 , 6) ;

System.out.println("\nArea of Rectangle : " + shape.getArea()) ;

System.out.println("Perimeter of Rectangle : " + shape.getPerimeter() + "\n") ;

shape = new RtTriangle(3 , 4) ;

System.out.println("Area of Right Triangle : " + shape.getArea()) ;

System.out.println("Perimeter of Right Triangle : " + shape.getPerimeter()+"\n") ;

shape = new Circle(6) ;

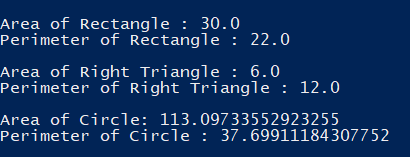
System.out.println("Area of Circle: " + shape.getArea()) ;

System.out.println("Perimeter of Circle : " + shape.getPerimeter()+"\n") ;

}

}

**OUTPUT**

****