Atividade Polimorfismos – Figuras Geométricas

Professor: Jonatas Fischer

Nome: Johnatas Félix Zuza

Matrícula: 422160

Q1)

**package** polimorfismo;

**public** **class** Main {

**public** **static** **void** main(String[] args) {

Torre t = **new** Torre();

t.addFormaGeometrica(**new** Quadrado(10));

t.addFormaGeometrica(**new** Triangulo(3,4,5));

t.addFormaGeometrica(**new** Retangulo(10, 15));

t.addFormaGeometrica(**new** Losangulo(6,12));

t.addFormaGeometrica(**new** Circulo(10));

System.***out***.println(t.calcularArea());

System.***out***.println(t.validarFormas());

}

}

**package** polimorfismo;

**public** **interface** ehFormaGeometricaValida {

**public** **boolean** formaValida();

}

**package** polimorfismo;

**public** **abstract** **class** FormaGeometrica **implements** ehFormaGeometricaValida {

**protected** **double** lado;

**public** **abstract** **double** calcularArea();

}

**package** polimorfismo;

**import** java.util.ArrayList;

**public** **class** Torre {

**private** ArrayList<FormaGeometrica> listaFormas;

**public** Torre() {

listaFormas = **new** ArrayList<FormaGeometrica>();

}

**public** **void** addFormaGeometrica(FormaGeometrica forma) {

listaFormas.add(forma);

}

**public** **double** calcularArea() {

**double** area = 0;

**for** (FormaGeometrica fm : listaFormas) {

area += fm.calcularArea();

}

**return** area;

}

**public** **boolean** validarFormas() {

**for** (FormaGeometrica formaGeometrica : listaFormas) {

**if**(!formaGeometrica.formaValida())

**return** **false**;

}

**return** **true**;

}

}

**package** polimorfismo;

**public** **class** Circulo **extends** FormaGeometrica {

**private** **double** pi = 3.14;

**private** **double** diametro;

**public** Circulo(**double** diametro) {

**this**.diametro = diametro;

}

@Override

**public** **boolean** formaValida() {

**if**(diametro <=0) {

System.***out***.println("Circulo invalido");

**return** **false**;

}**else**

**return** **true**;

}

@Override

**public** **double** calcularArea() {

**return** pi \* ((diametro/2)\*(diametro/2));

}

}

**package** polimorfismo;

**public** **class** Losangulo **extends** FormaGeometrica {

**private** **double** diagonal1;

**private** **double** diagonal2;

**public** Losangulo(**double** diagonal1, **double** diagonal2) {

**this**.diagonal1 = diagonal1;

**this**.diagonal2 = diagonal2;

}

**public** **double** calcularArea() {

**return** (diagonal1 \* diagonal2) \* 0.5 ;

}

@Override

**public** **boolean** formaValida() {

**if**(diagonal1 == diagonal2) {

System.***out***.println("Losangulo invalido");

**return** **false**;

}**else** **if**(diagonal1 == 0 || diagonal2 ==0) {

System.***out***.println("Losangulo invalido");

**return** **false**;

}**else**

**return** **true**;

}

}

**package** polimorfismo;

**public** **class** Quadrado **extends** FormaGeometrica{

**public** Quadrado(**int** lado) {

**this**.lado=lado;

}

**public** **double** calcularArea() {

**return** lado\*lado;

}

@Override

**public** **boolean** formaValida() {

**if**(lado <=0) {

System.***out***.println("Quadrado inválido");

**return** **false**;

}

**return** **true**;

}

}

**package** polimorfismo;

**public** **class** Retangulo **extends** FormaGeometrica {

**private** **double** segundoLado;

**public** Retangulo(**int** lado, **int** segundoLado) {

**this**.lado=lado;

**this**.segundoLado=segundoLado;

}

**public** **double** calcularArea() {

**return** lado \* segundoLado;

}

**public** **boolean** formaValida() {

**if**(**this**.lado == **this**.segundoLado) {

System.***out***.println("Retangulo invalido");

**return** **false**;

}

**else** **if** (lado <= 0 || segundoLado <=0) {

System.***out***.println("Retangulo invalido");

**return** **false**;

}

**return** **true**;

}

}

**package** polimorfismo;

**public** **class** Triangulo **extends** FormaGeometrica {

**private** **double** lado2;

**private** **double** lado3;

**public** Triangulo(**double** lado, **double** lado2, **double** lado3) {

**this**.lado = lado;

**this**.lado2 = lado2;

**this**.lado3 = lado3;

}

**public** **double** calcularArea() {

**double** p = (lado + lado2 + lado3) / 2;

**double** area = Math.*sqrt*(p \* (p - lado) \* (p - lado2) \* (p - lado3));

**return** area;

}

**public** **boolean** formaValida() {

**if**(lado <= 0 || lado2 <=0 || lado3 <= 0) {

System.***out***.println("Triangulo invalido");

**return** **false**;

}

**return** **true**;

}

}