INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO

CARRERA:

INGENIERÍA EN SISTEMAS COMPUTACIONALES

Materia:

Paradigmas de programación

Profesor: García Floriano Andrés

Alumno:

Escobar Rodriguez Alfonso 3CV1

Fecha de entrega: 23 de abril del 2024.

ESCOM

Código

Python

```
import math

# FIGURA
class Figure:
    def __init__(self, color):
        self.color = color

    def getColor(self):
        return self.color

    def perimeter(self):
        pass

    def area(self):
        pass

# TRIANGULO
class Triangle(Figure):
    def __init__(self, color, side1, side2, side3):
        super().__init__(color)
        self.side1 = side1
        self.side2 = side2
        self.side3 = side3

    def perimeter(self):
        return self.side1 + self.side2 + self.side3

    def area(self):
        s = self.perimeter() / 2
        return math.sqrt(s * (s - self.side1) * (s - self.side2) * (s -
self.side3))

# CIRCULO
class Circle(Figure):
    def __init__(self, color, radius):
        super().__init__(color)
        self.radius = radius

    def perimeter(self):
        return 2 * math.pi * self.radius

    def area(self):
        return math.pi * self.radius ** 2

# RECTANGULO
class Rectangle(Figure):
    def __init__(self, color, width, height):
        super().__init__(color)
        self.width = width
        self.height = height

    def perimeter(self):


      return 2 * (self.width + self.height)

    def area(self):
        return self.width * self.height

# HEXAGONO
class Hexagon(Figure):
    def __init__(self, color, side):
        super().__init__(color)
        self.side = side

    def perimeter(self):
        return 6 * self.side

    def area(self):
        return (3 * math.sqrt(3) * self.side ** 2) / 2

def demonstrate_polymorphism(figures):
    for figure in figures:
        print(f"Color: {figure.getColor()}")
        print(f"Perimeter: {figure.perimeter()}")
        print(f"Area: {figure.area()}\n")

triangle = Triangle("Red", 3, 4, 5)
circle = Circle("Blue", 7)
rectangle = Rectangle("Green", 5, 10)
hexagon = Hexagon("Yellow", 6)

figures = [triangle, circle, rectangle, hexagon]

demonstrate_polymorphism(figures)
```

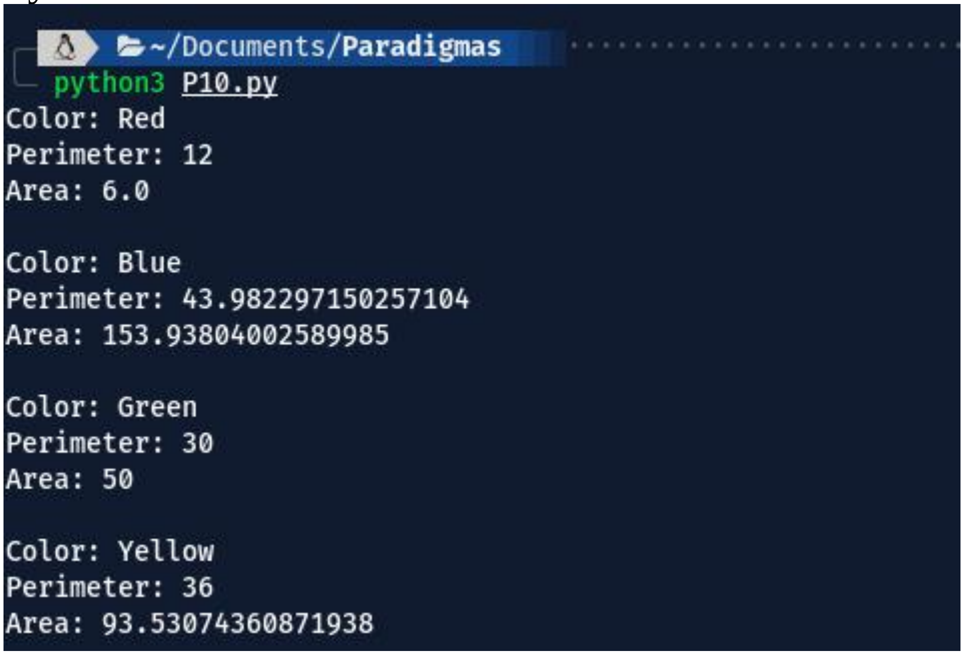
JAVA

```
import java.lang.Math;

// Figura
abstract class Figure {
    private String color;

    public Figure(String color) {
        this.color = color;
    }

    public String getColor() {
        return color;
    }
    public abstract double perimeter();
    public abstract double area();
}
```

```java
// Triángulo
class Triangle extends Figure {
    private double side1;
    private double side2;
    private double side3;

    public Triangle(String color, double side1, double side2, double side3) {
        super(color);
        this.side1 = side1;
        this.side2 = side2;
        this.side3 = side3;
    }

    @Override
    public double perimeter() {
        return side1 + side2 + side3;
    }

    @Override
    public double area() {
        double s = perimeter() / 2;
        return Math.sqrt(s * (s - side1) * (s - side2) * (s - side3));
    }
}

// Círculo
class Circle extends Figure {
    private double radius;

    public Circle(String color, double radius) {
        super(color);
        this.radius = radius;
    }

    @Override
    public double perimeter() {
        return 2 * Math.PI * radius;
    }

    @Override
    public double area() {
        return Math.PI * Math.pow(radius, 2);
    }
}

// Rectángulo
class Rectangle extends Figure {
    private double width;
    private double height;

    public Rectangle(String color, double width, double height) {
        super(color);
        this.width = width;
        this.height = height;
    }

    @Override
    public double perimeter() {
        return 2 * (width + height);
    }
    @Override
    public double area() {
        return width * height;
    }
}

// Hexágono
class Hexagon extends Figure {
    private double side;

    public Hexagon(String color, double side) {
        super(color);
        this.side = side;
    }

    @Override
    public double perimeter() {
        return 6 * side;
    }

    @Override
    public double area() {
        return (3 * Math.sqrt(3) * Math.pow(side, 2)) / 2;
    }
}

public class Main {
    public static void demonstratePolymorphism(Figure[] figures) {
        for (Figure figure : figures) {
            System.out.println("Color: " + figure.getColor());
            System.out.println("Perimeter: " + figure.perimeter());
            System.out.println("Area: " + figure.area());
            System.out.println();
        }
    }

    public static void main(String[] args) {
        Figure triangle = new Triangle("Red", 3, 4, 5);
        Figure circle = new Circle("Blue", 7);
        Figure rectangle = new Rectangle("Green", 5, 10);
        Figure hexagon = new Hexagon("Yellow", 6);

        Figure[] figures = {triangle, circle, rectangle, hexagon};

        demonstratePolymorphism(figures);
    }
}
```

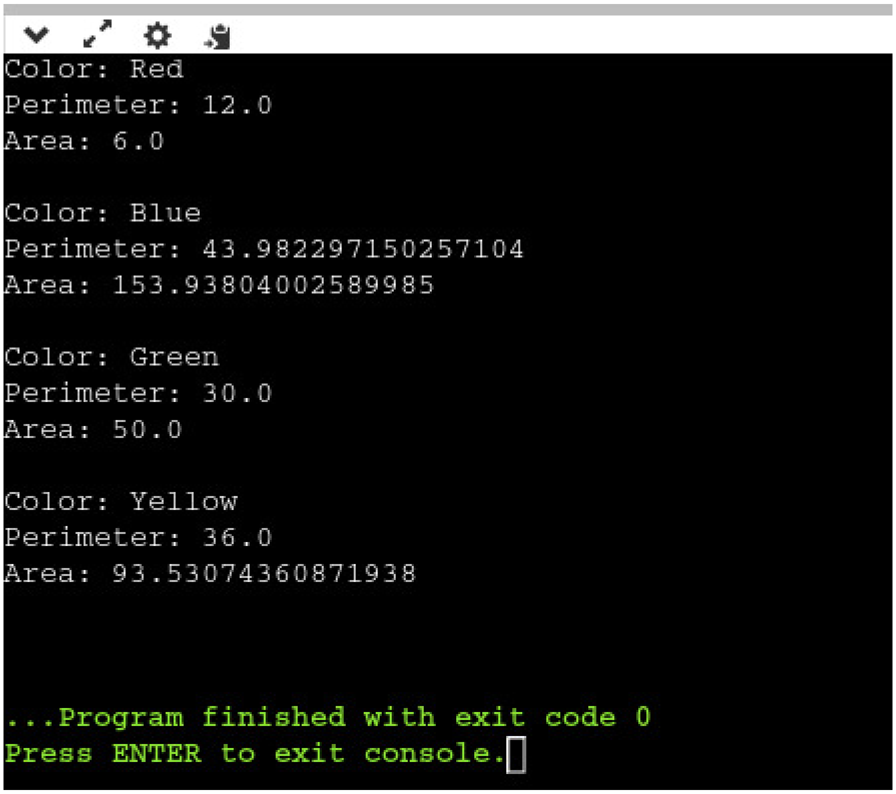
Prueba de Validez

Python



```
~/Documents/Paradigmas
python3 P10.py
Color: Red
Perimeter: 12
Area: 6.0

Color: Blue
Perimeter: 43.982297150257104
Area: 153.93804002589985

Color: Green
Perimeter: 30
Area: 50

Color: Yellow
Perimeter: 36
Area: 93.53074360871938
```

JAVA



```
Color: Red
Perimeter: 12.0
Area: 6.0

Color: Blue
Perimeter: 43.982297150257104
Area: 153.93804002589985

Color: Green
Perimeter: 30.0
Area: 50.0

Color: Yellow
Perimeter: 36.0
Area: 93.53074360871938



...Program finished with exit code 0
Press ENTER to exit console.
```