Name : **R Anush** **Date : 1/10/2023**
Student Code : **AF0336714**
Batch Code : **Java_ANP-C6315**

Lab Assignment-16

Q1: .Create an abstract class Shape with an abstract method calculateArea(). Implement two subclasses, Circle and Rectangle, which inherit from Shape and provide their own implementations of calculateArea().

Write a program to calculate and print the areas of a circle and a rectangle.

Input:

```
package CoreJava;

// Abstract Shape class
abstract class Shape {
    // Abstract method to calculate the area
    public abstract double calculateArea();
}

// Circle subclass
class Circle extends Shape {
    private double radius;

    public Circle(double radius) {
        this.radius = radius;
    }

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

// Rectangle subclass
class Rectangle extends Shape {
```

```
    private double length;
    private double width;

    public Rectangle(double length, double width) {
        this.length = length;
        this.width = width;
    }

    @Override
    public double calculateArea() {
        return length * width;
    }
}

public class Main {
    public static void main(String[] args) {
        // Create a circle with a radius of 5 units
        Circle circle = new Circle(5.0);
        // Calculate and print the area of the circle
        System.out.println("Area of the Circle: " + circle.calculateArea());

        // Create a rectangle with length 4 units and width 6 units
        Rectangle rectangle = new Rectangle(4.0, 6.0);
        // Calculate and print the area of the rectangle
        System.out.println("Area of the Rectangle: " + rectangle.calculateArea());
    }
}
```

Output:

Area of the Circle: 78.53981633974483

Area of the Rectangle: 24.0

Q2: Write a Java program that demonstrates method overriding by creating a superclass called Animal and two subclasses called Dog and Cat.

● The Animal class should have a method called makeSound(), which simply prints "The animal makes a sound."

● The Dog and Cat classes should override this method to print "TheCat/The dog meows/barks" respectively.

● The program should allow the user to create and display objects of each class. [Hint:Use multilevel inheritance]

Input:

```
package CoreJava;

public class Animal {

        public void makeSound() {
       System.out.println("The animal makes a sound.");
    }
}

class Dog extends Animal {
    @Override
    public void makeSound() {
        System.out.println("The dog barks.");
    }
}

class Cat extends Animal {
    @Override
    public void makeSound() {
        System.out.println("The cat meows.");
    }
}

public class Main {
    public static void main(String[] args) {
        Animal animal = new Animal();
        Dog dog = new Dog();
```

```
        Cat cat = new Cat();

        System.out.println("Animal Sound:");
        animal.makeSound();

        System.out.println("\nDog Sound:");
        dog.makeSound();

        System.out.println("\nCat Sound:");
        cat.makeSound();
    }
}
```

Output:

Animal Sound:

The animal makes a sound.

Dog Sound:

The dog barks.

Cat Sound:

The cat meows.