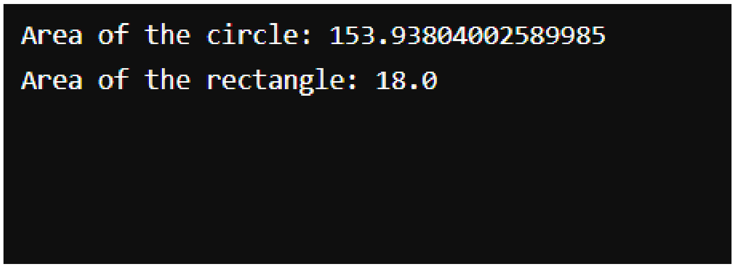
Assignment-1

1. 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.

Program:

```
abstract class Shape {

 // Abstract method public abstract

 double calculateArea();

 }

// Subclass for Circle

class Circle extends Shape {

private double radius;

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

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


}

// Subclass

class Rectangle extends Shape {

private double width;

private double height;

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

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


}

// Main class to run the program

public class Main {

public static void main(String[] args) {

Shape circle  = new Circle(7.0);

        Shape rectangle = new Rectangle(2.0, 9.0);

   System.out.println("Area of the circle: " + circle.calculateArea());
    System.out.println("Area of the rectangle: " + rectangle.calculateArea());
}


}
```

Output:



```
Area of the circle: 153.93804002589985
Area of the rectangle: 18.0
```

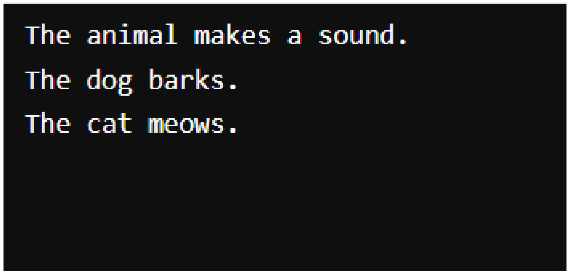
2 Write a Java program that demonstrates method overriding by creating a superclass called Animal and two subclasses called Dog and Cat.

- 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.

Program:

```
class Animal {
    // Method to be overridden
    public void makeSound() {
        System.out.println("The animal makes a sound.");
    }
}

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

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

// Main class
public class Main {
    public static void main(String[] args) {
        Animal animal = new Animal();
        animal.makeSound(); // Output: The animal makes a sound.

        Dog dog = new Dog();
        dog.makeSound(); // Output: The dog barks.

        Cat cat = new Cat();
        cat.makeSound(); // Output: The cat meows.
    }
}
```

Output:



```
The animal makes a sound.
The dog barks.
The cat meows.
```