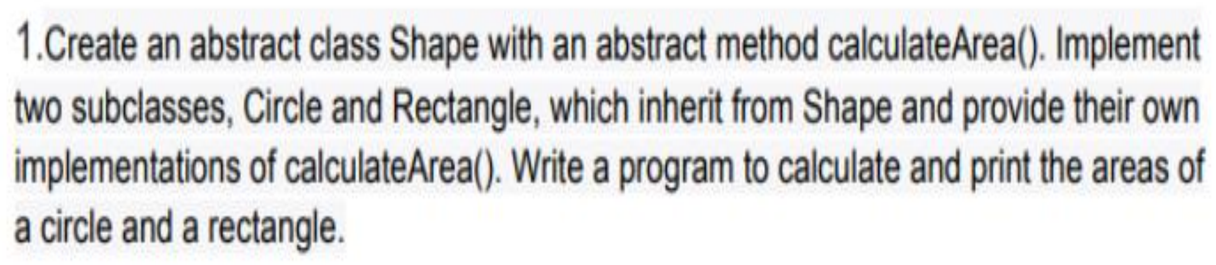
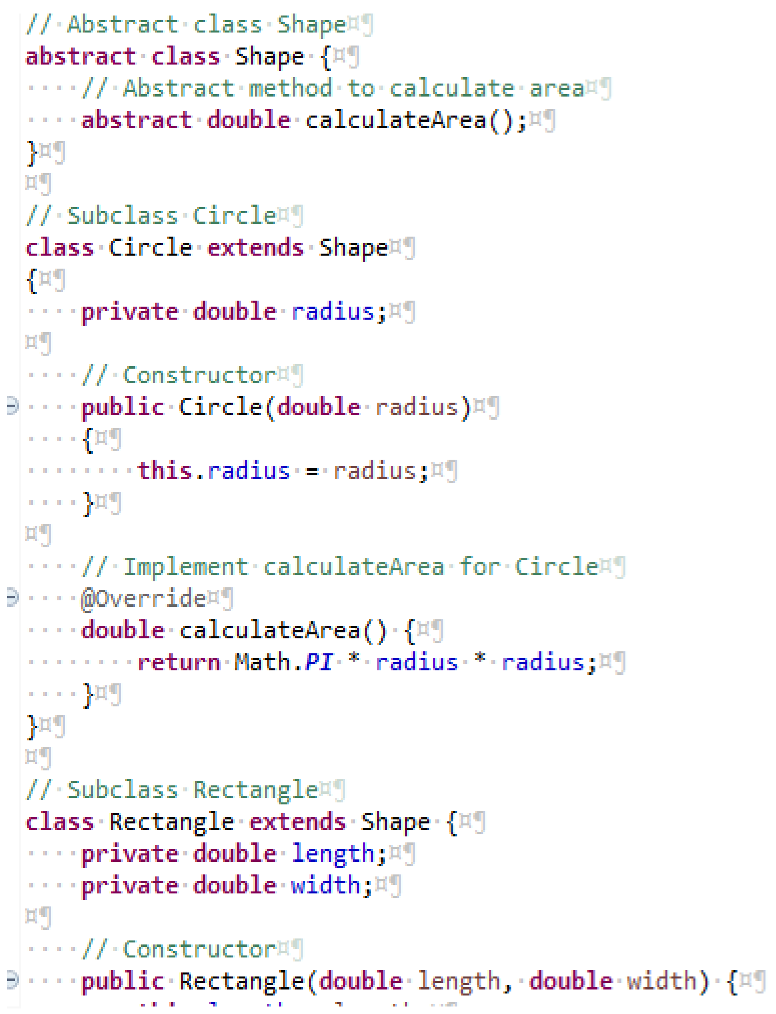
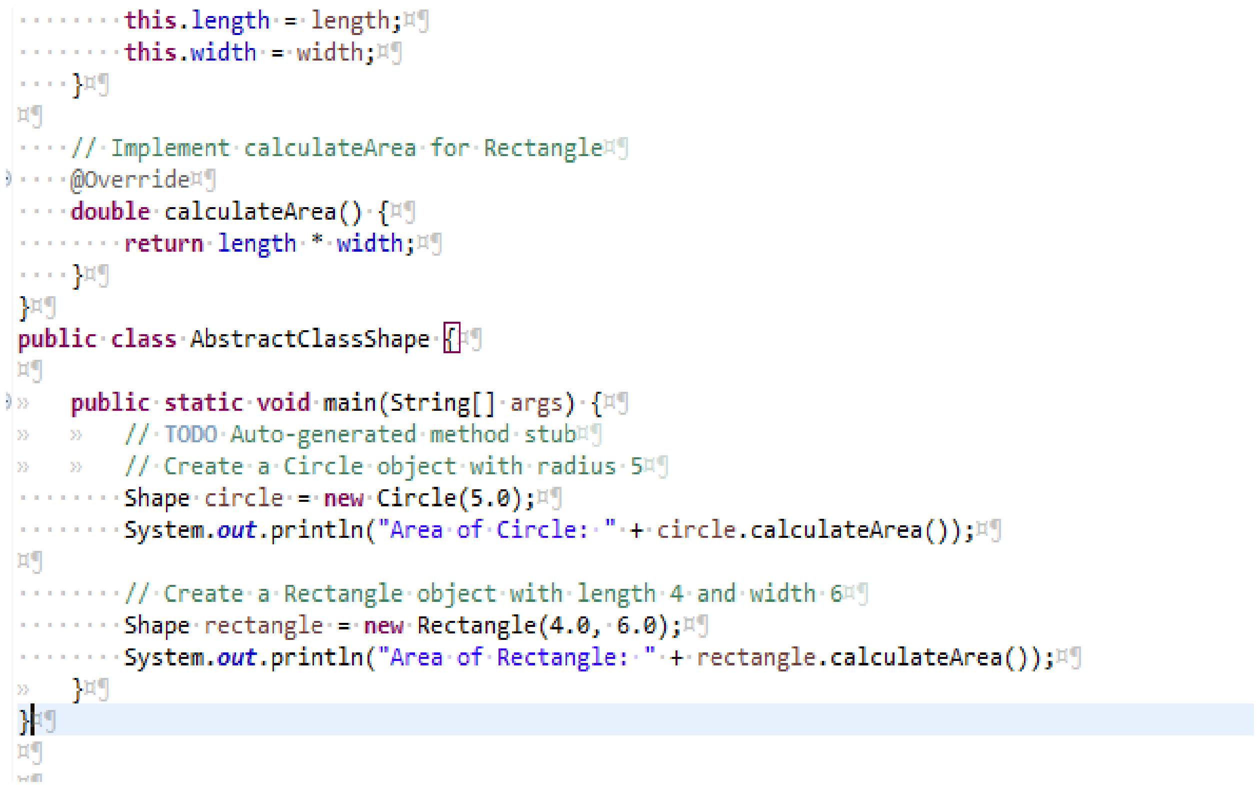
Lab. – 8

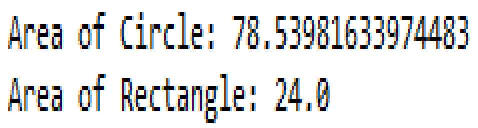

1.Create an abstract class Shape with an abstract method calculateArea(). Implement two subclasses, Circle and Rectangle, which inherit from Shape and provide their own implementations of calculateArea(). Write a program to calculate and print the areas of a circle and a rectangle.


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

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

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

    // Implement calculateArea for Circle
    @Override
    double calculateArea() {
        return Math.PI * radius * radius;
    }
}

// Subclass Rectangle
class Rectangle extends Shape {
    private double length;
    private double width;

    // Constructor
    public Rectangle(double length, double width) {
```



```java
        this.length = length;
        this.width = width;
    }

    // Implement calculateArea for Rectangle
    @Override
    double calculateArea() {
        return length * width;
    }
}
public class AbstractClassShape {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        // Create a Circle object with radius 5
        Shape circle = new Circle(5.0);
        System.out.println("Area of Circle: " + circle.calculateArea());

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

Output

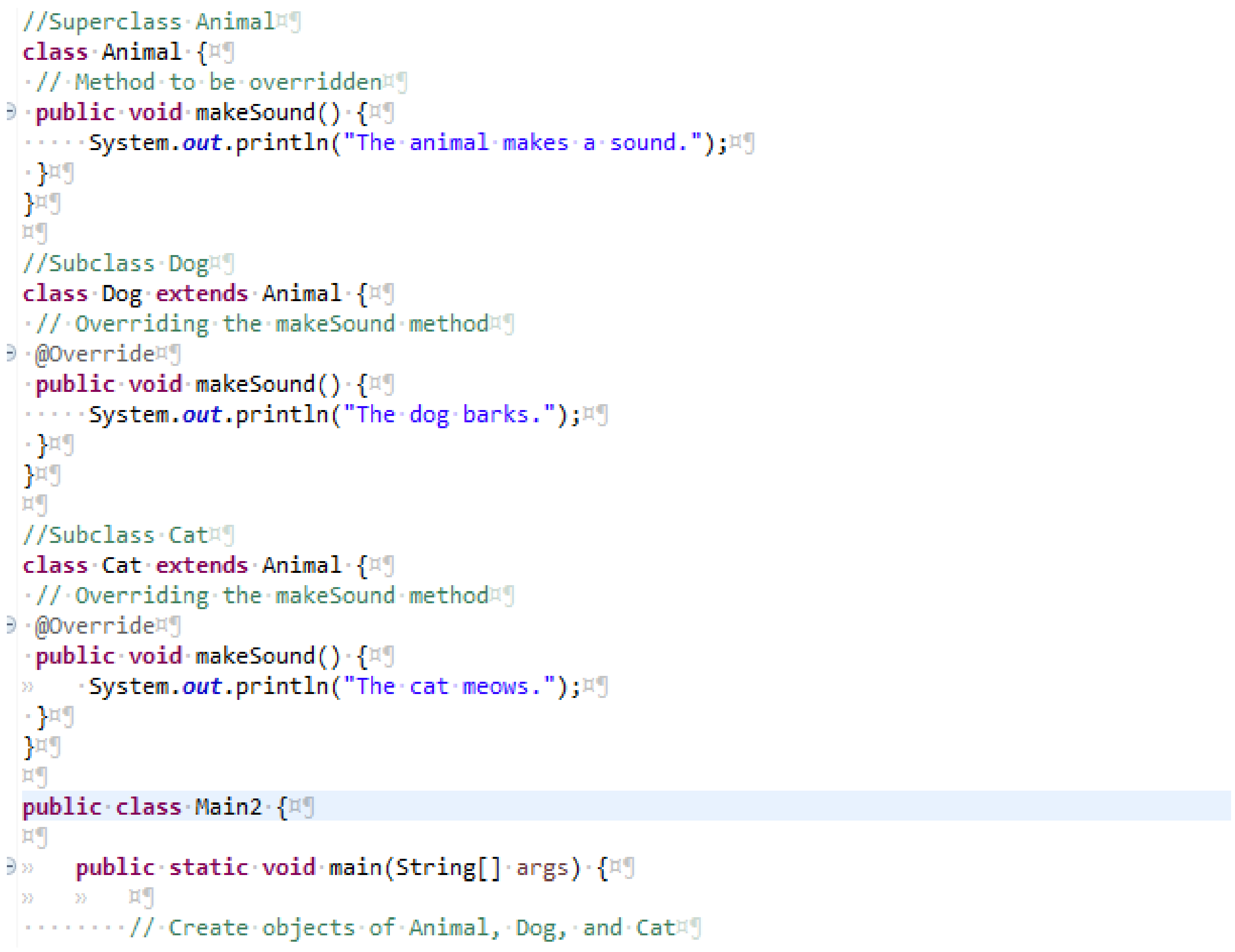


```
Area of Circle: 78.53981633974483
Area of Rectangle: 24.0
```

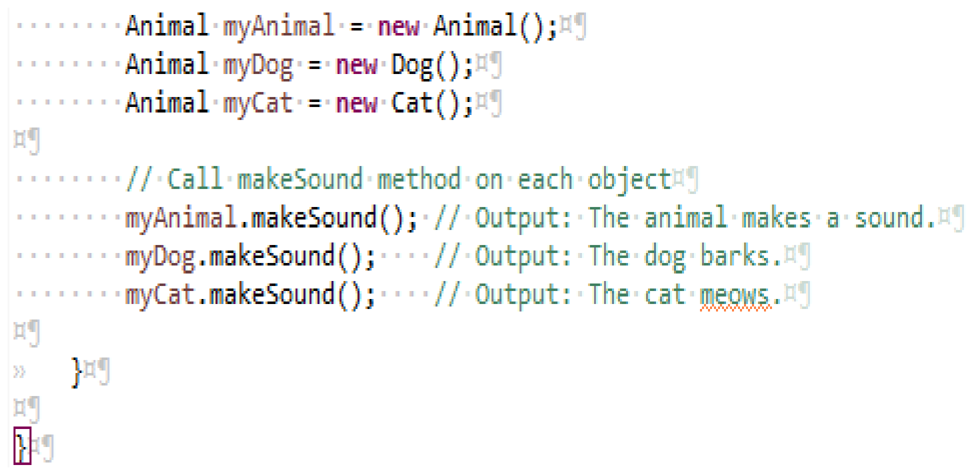
2. Write a Java program that demonstrates method overriding by creating a superclass called Animal and two subclasses called Dog and Cat.

- 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.

[Hint:Use multilevel inheritance]

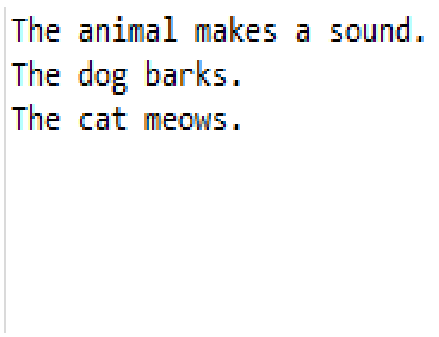


```java
//Superclass Animal
class Animal {
 // Method to be overridden
 public void makeSound() {
     System.out.println("The animal makes a sound.");
 }
}

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

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

public class Main2 {

    public static void main(String[] args) {
        
        // Create objects of Animal, Dog, and Cat
```



```java
        Animal myAnimal = new Animal();
        Animal myDog = new Dog();
        Animal myCat = new Cat();

        // Call makeSound method on each object
        myAnimal.makeSound(); // Output: The animal makes a sound.
        myDog.makeSound();    // Output: The dog barks.
        myCat.makeSound();    // Output: The cat meows.

    }

}
```

Output



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