# CLASS TEST 4

1. **Create a base class called Shape with a virtual function area(). Derive two classes Rectangle and Circle from the base class. Implement the area() function for each class.**

## PROGRAM:

```
class Shape {
    public double area() {
        return 0;
    }
}
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 area() {
        return length * width;
    }
}
class Circle extends Shape {
    private double radius;
```

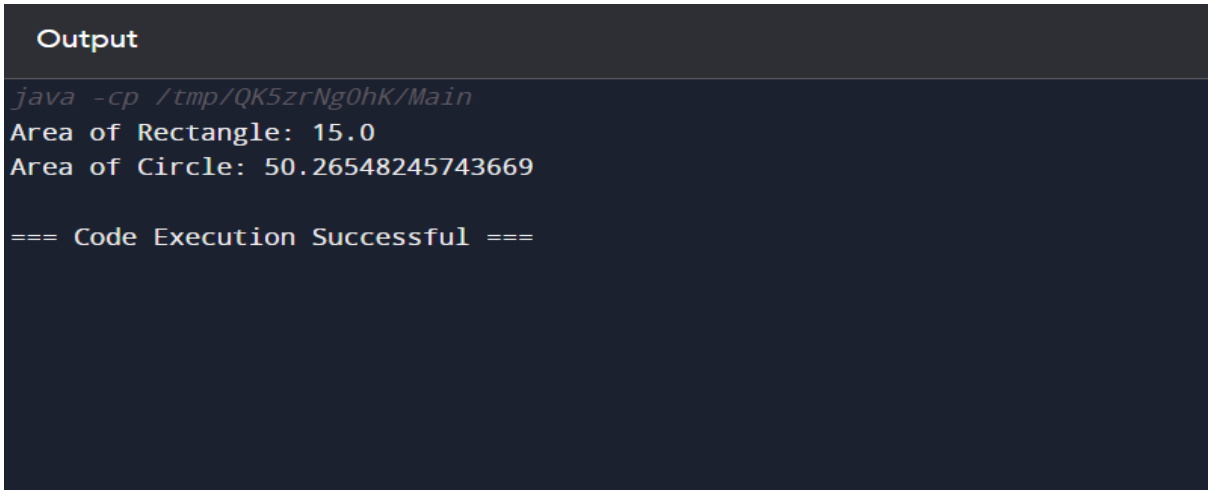
```
    public Circle(double radius) {
        this.radius = radius;
    }

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

public class Main {
    public static void main(String[] args) {
        Rectangle rectangle = new Rectangle(5, 3);
        Circle circle = new Circle(4);

        System.out.println("Area of Rectangle: " + rectangle.area());
        System.out.println("Area of Circle: " + circle.area());
    }
}
```

**OUTPUT:**



2. **Create a base class called Animal with a virtual function speak(). Derive two classes Cat and Dog from the base class. Implement the speak() function for each class.**

## PROGRAM:

```
class Animal {
    public void speak() {
        System.out.println("The animal speaks");
    }
}

class Cat extends Animal {
    @Override
    public void speak() {
        System.out.println("Meow");
    }
}

class Dog extends Animal {
    @Override
    public void speak() {
        System.out.println("Woof");
    }
}

public class Main {
```

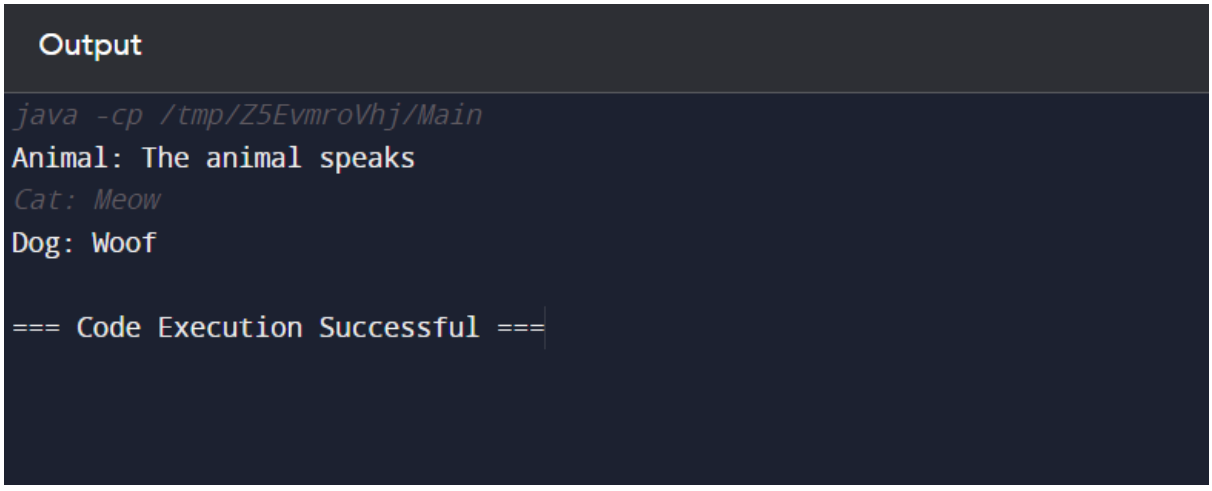
```
    public static void main(String[] args) {
        Animal animal = new Animal();
        Cat cat = new Cat();
        Dog dog = new Dog();

        System.out.print("Animal: ");
        animal.speak();

        System.out.print("Cat: ");
        cat.speak();

        System.out.print("Dog: ");
        dog.speak();
    }
}
```

**OUTPUT:**




3. **Create a base class called Employee with a virtual function calculatePay(). Derive two classes Manager and Engineer from the base class. Implement the calculatePay() function for each class.**

**PROGRAM:**

```
class Employee {
    public void calculatePay() {
        System.out.println("Calculating pay for Employee");
    }
}

class Manager extends Employee {
    @Override
    public void calculatePay() {
        System.out.println("Calculating pay for Manager");
    }
}

class Engineer extends Employee {
    @Override
    public void calculatePay() {
        System.out.println("Calculating pay for Engineer");
    }
}

public class Main {
    public static void main(String[] args) {
        Employee emp1 = new Manager();
        Employee emp2 = new Engineer();
```

```
        emp1.calculatePay();
        emp2.calculatePay();
    }
}
```

## OUTPUT: