Inheritance

● Write a Java program to create a class called Shape with a method called getArea().

● Create a subclass called Circle and create a constructor that takes the value of radius(int) as input parameter.

● Override the getArea() method.

● Create a class called square that takes an attribute length. Create a constructor that takes length as input.

● Override the getArea() method.

● Create a subclass of Shape called Rectangle that takes width and height as input to the constructor.

● Override the getArea() method to calculate the area of a rectangle. Instantiate and call the getArea() method.

```
class Shape {
    // Method to calculate and return the area
    public double getArea() {
        return 0.0; // Default implementation, to be overridden by subclasses
    }
}

class Circle extends Shape {
    private int radius;

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

    // Override getArea() method for Circle
    @Override
    public double getArea() {
```

```
        return Math.PI * radius * radius;
    }
}

class Square extends Shape {
    private int length;

    // Constructor
    public Square(int length) {
        this.length = length;
    }

    // Override getArea() method for Square
    @Override
    public double getArea() {
        return length * length;
    }
}

class Rectangle extends Shape {
    private int width;
    private int height;

    // Constructor
    public Rectangle(int width, int height) {
        this.width = width;
        this.height = height;
    }

    // Override getArea() method for Rectangle
    @Override
    public double getArea() {
        return width * height;
    }
}

public class Main {
    public static void main(String[] args) {
        // Instantiate Circle and calculate its area
        Circle circle = new Circle(5);
        System.out.println("Area of Circle: " + circle.getArea());

        // Instantiate Square and calculate its area
        Square square = new Square(4);
```

```
        System.out.println("Area of Square: " + square.getArea());

        // Instantiate Rectangle and calculate its area
        Rectangle rectangle = new Rectangle(3, 6);
        System.out.println("Area of Rectangle: " + rectangle.getArea());
    }
}
```

● Write a Java program to create Vehiclae class & extends subclasses which shows inheritance

```
public class Vehicle {

        String make;
        String model;
        int year;
        int maximumSpeed;

        void drive() {
                System.out.println(make + " " + model + " is driving.");
        }

        // Constructor for Vehicle
        public Vehicle(String make, String model, int year, int maximumSpeed) {
                super();
                this.make = make;
                this.model = model;
                this.year = year;
                this.maximumSpeed = maximumSpeed;
        }

}

class Car extends Vehicle{

        // Constructor for Car
        public Car(String make, String model, int year, int maximumSpeed) {
                super(make, model, year, maximumSpeed);
                // TODO Auto-generated constructor stub
        }
```

```
        @Override
    void drive() {
         System.out.println(make + " " + model + " Car is driving.");
        }

}

class Bike extends Vehicle {
    // Constructor for Bike
    public Bike(String make, String model, int year, int maximumSpeed) {
        super(make, model, year, maximumSpeed);
    }

    // Override the drive method in Bike class
    @Override
    void drive() {
        System.out.println(make + " " + model + " Bike is driving.");
    }
}
```

By - Divya Parihar