Assignment –1

- Write a Java program to create a class called Vehicle with a method called drive().
- Vehicle should have attributes such as make (String), model (String), year (int) and maximumSpeed (int).
- Create a constructor in Vehicle with all fields as constructor parameters.
- Create a subclass called Car and override constructor. Call super()..

Write a function that overrides the drive() method to print (make + " " + model +" Car is driving".)

- Also create another subclass Bike extending the vehicle class. .

Override the drive() method to print (make + " " + model + " Bike is driving".)

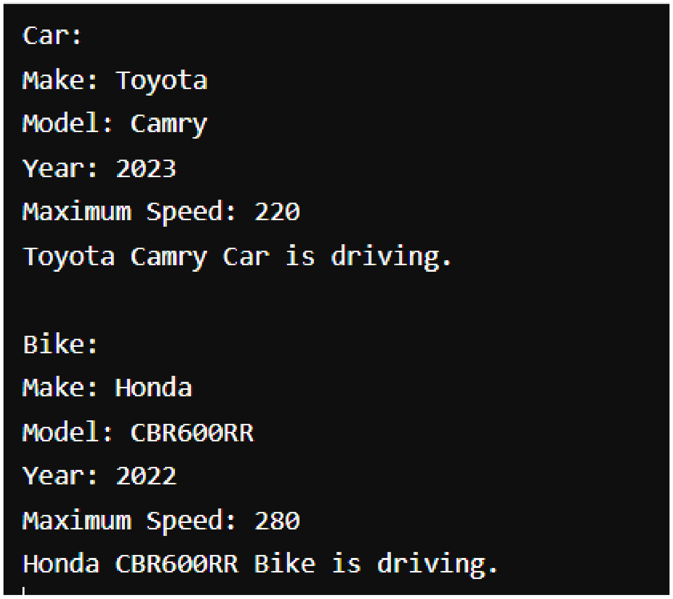
- Instantiate both Bike and Car class. Print their attributes.

Program:

```
class Vehicle {
    String make;
    String model;
    int year;
    int maximumSpeed;

// Constructor to initialize Vehicle attributes
    public Vehicle(String make, String model, int year, int maximumSpeed) {
        this.make = make;
        this.model = model;
        this.year = year;
        this.maximumSpeed = maximumSpeed;
    }
// A method to be overridden by subclasses
    public void drive() {
        System.out.println("Vehicle is driving.");
    }
}

// Child class: Car (inherits from Vehicle)
class Car extends Vehicle {
// Constructor that uses the parent class constructor via super()
```

```
    public Car(String make, String model, int year, int maximumSpeed) {
        super(make, model, year, maximumSpeed);
    }
// Overriding the drive method
    @Override
    public void drive() {
        System.out.println(make + " " + model + " Car is driving.");
    }
}

// Child class: Bike (inherits from Vehicle)
class Bike extends Vehicle {
    public Bike(String make, String model, int year, int maximumSpeed) {
        super(make, model, year, maximumSpeed);
    }

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

// Main class to demonstrate
public class Main {
    public static void main(String[] args) {
        Car car = new Car("Toyota", "Camry", 2023, 220);
        Bike bike = new Bike("Honda", "CBR600RR", 2022, 280);

// Print attributes and call the overridden drive methods
        System.out.println("Car:");
        System.out.println("Make: " + car.make);
        System.out.println("Model: " + car.model);
        System.out.println("Year: " + car.year);
        System.out.println("Maximum Speed: " + car.maximumSpeed);
        car.drive(); //Calls Car's drive() method

        System.out.println("\nBike:");
        System.out.println("Make: " + bike.make);
        System.out.println("Model: " + bike.model);
        System.out.println("Year: " + bike.year);
        System.out.println("Maximum Speed: " + bike.maximumSpeed);
        bike.drive();// Calls Bike's drive() method
    }
}
```

Output:



```
Car:
Make: Toyota
Model: Camry
Year: 2023
Maximum Speed: 220
Toyota Camry Car is driving.

Bike:
Make: Honda
Model: CBR600RR
Year: 2022
Maximum Speed: 280
Honda CBR600RR Bike is driving.
```

Assignment –2

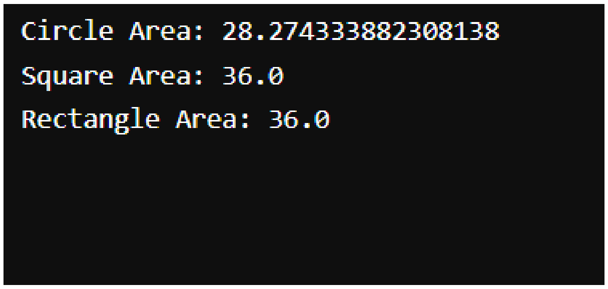
- Write a Java program to create a class called Shape with a methi getArea().
- Create a subclass called Circle and create a constructor that take of radius(int) as input parameter.
- Override the getArea() method.
- Create a class called square that takes an attribute length. Creat constructor that takes length as input.
- Override the getArea() method.
- Create a subclass of Shape called Rectangle that takes width an input to the constructor.
- Override the getArea() method to calculate the area of a rectangl Instantiate and call getArea() method.

Program:

```
class Shape {

// Abstract method to calculate area

 public double getArea() {

 return 0.0;

 // Default implementation, can be overridden

 }

 }

 // Constructor to initialize radius

class Circle extends Shape {

private int radius;

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

```
// Override the getArea() method

@Override

public double getArea() {

 return Math.PI * radius * radius;

}

}

// Subclass: Square (inherits from Shape)

 class Square extends Shape {

 private int side;

public Square(int side) {
    this.side = side;
}


// Constructor to initialize length

@Override

public double getArea() {

 return side * side;

 }

 }

// Subclass: Rectangle (inherits from Shape)

class Rectangle extends Shape {

private int width;

private int height;

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

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


}

// Main class to test the functionality

public class Main {

public static void main(String[] args) {

 // Create instances of Circle, Square, and Rectangle

Circle circle = new Circle(3); // Circle with radius 3

 Square square = new Square(6); // Square with side length 6

Rectangle rectangle = new Rectangle(4, 9);// Rectangle with width 4 and height 9

// Call getArea() and print the results

System.out.println("Circle Area: " + circle.getArea());

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

System.out.println("Rectangle Area: " + rectangle.getArea());

}

}
```

Output:



```
Circle Area: 28.274333882308138
Square Area: 36.0
Rectangle Area: 36.0
```