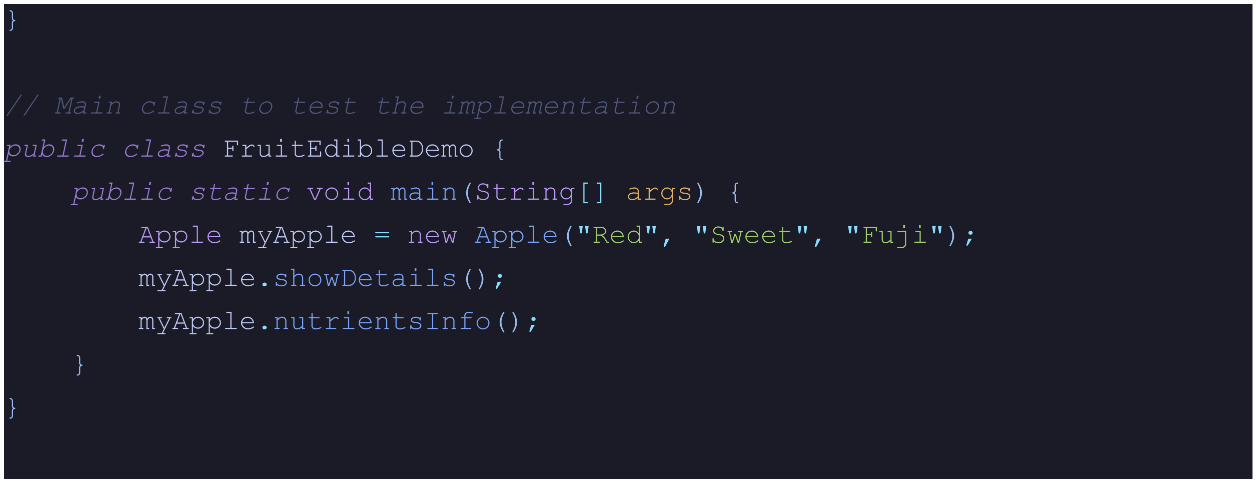
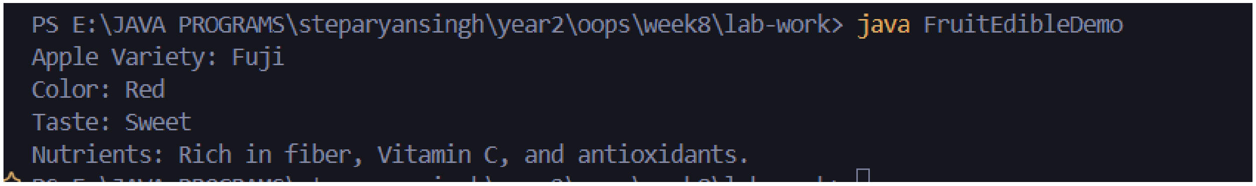
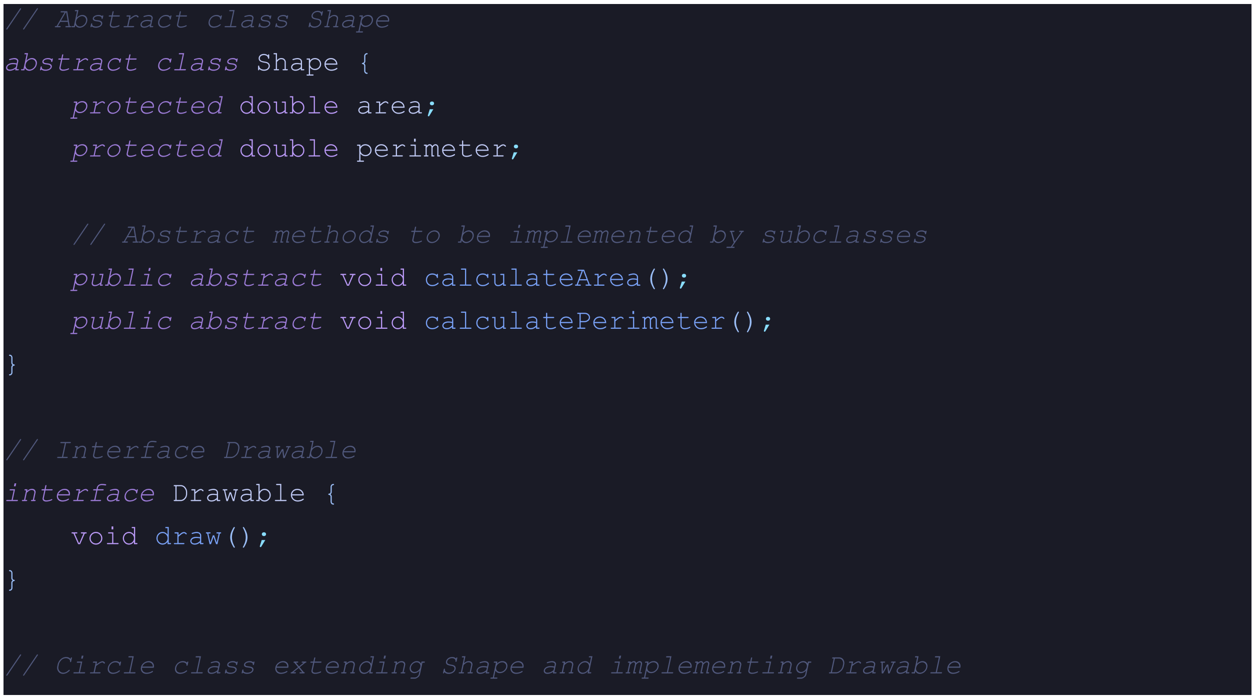
```java
// Abstract class Fruit
abstract class Fruit {
    protected String color;
    protected String taste;

    public Fruit(String color, String taste) {
        this.color = color;
        this.taste = taste;
    }

    // Abstract method to be implemented by subclasses
    public abstract void showDetails();
}

// Interface Edible
interface Edible {
    void nutrientsInfo();
}

// Apple class extending Fruit and implementing Edible
class Apple extends Fruit implements Edible {
    private String variety;

    public Apple(String color, String taste, String variety) {
        super(color, taste);
        this.variety = variety;
    }

    @Override
    public void showDetails() {
        System.out.println("Apple Variety: " + variety);
        System.out.println("Color: " + color);
        System.out.println("Taste: " + taste);
    }

    @Override
    public void nutrientsInfo() {
        System.out.println("Nutrients: Rich in fiber, Vitamin C, and
antioxidants.");
    }
```



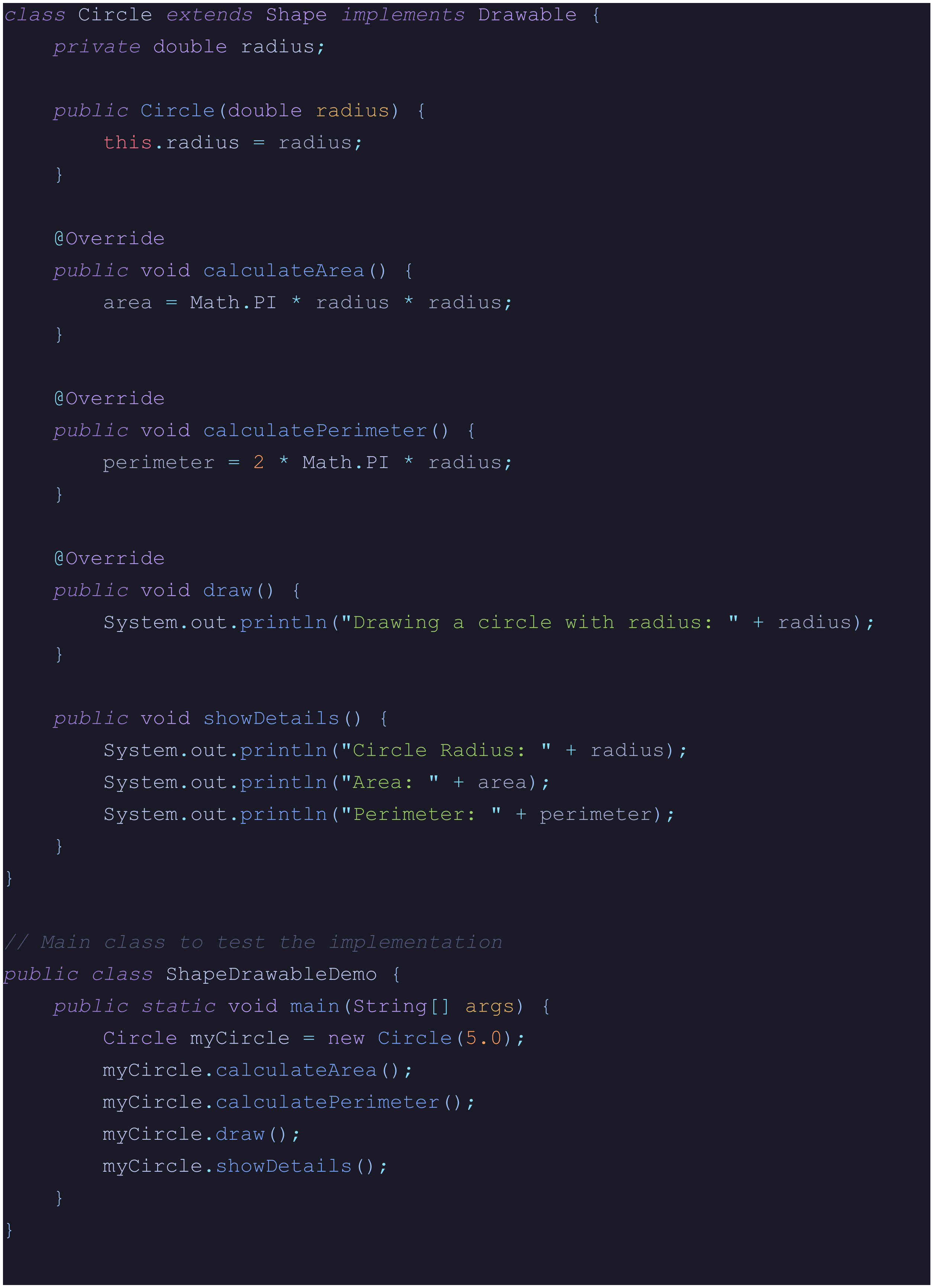
```java
}

// Main class to test the implementation
public class FruitEdibleDemo {
    public static void main(String[] args) {
        Apple myApple = new Apple("Red", "Sweet", "Fuji");
        myApple.showDetails();
        myApple.nutrientsInfo();
    }
}
```



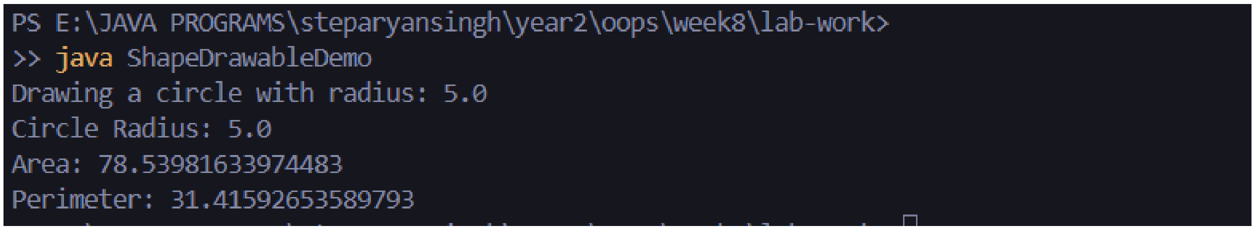
```
PS E:\JAVA PROGRAMS\steparyansingh\year2\oops\week8\lab-work> java FruitEdibleDemo
Apple Variety: Fuji
Color: Red
Taste: Sweet
Nutrients: Rich in fiber, Vitamin C, and antioxidants.
```



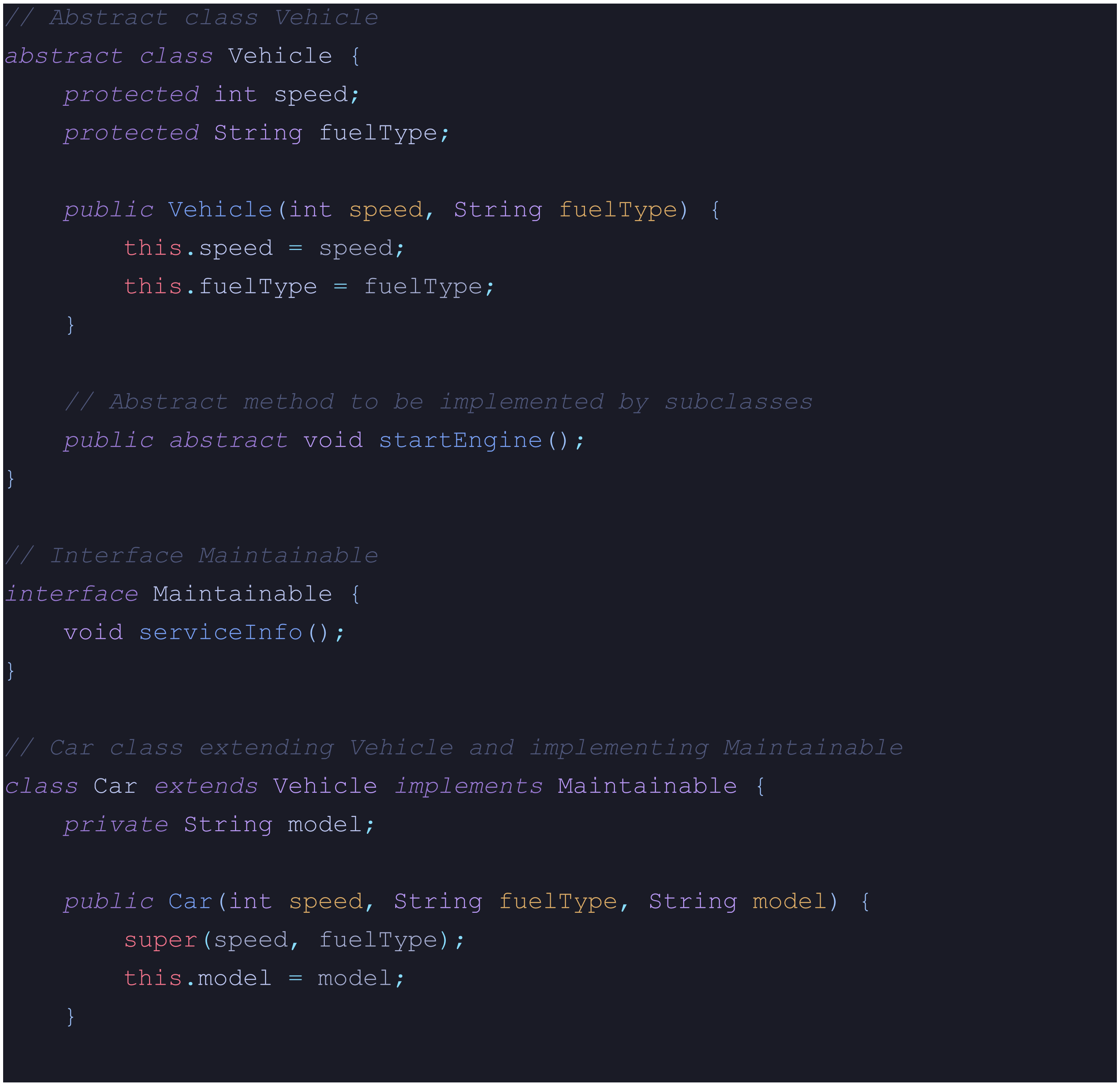
```java
// Abstract class Shape
abstract class Shape {
    protected double area;
    protected double perimeter;

    // Abstract methods to be implemented by subclasses
    public abstract void calculateArea();
    public abstract void calculatePerimeter();
}

// Interface Drawable
interface Drawable {
    void draw();
}

// Circle class extending Shape and implementing Drawable
```



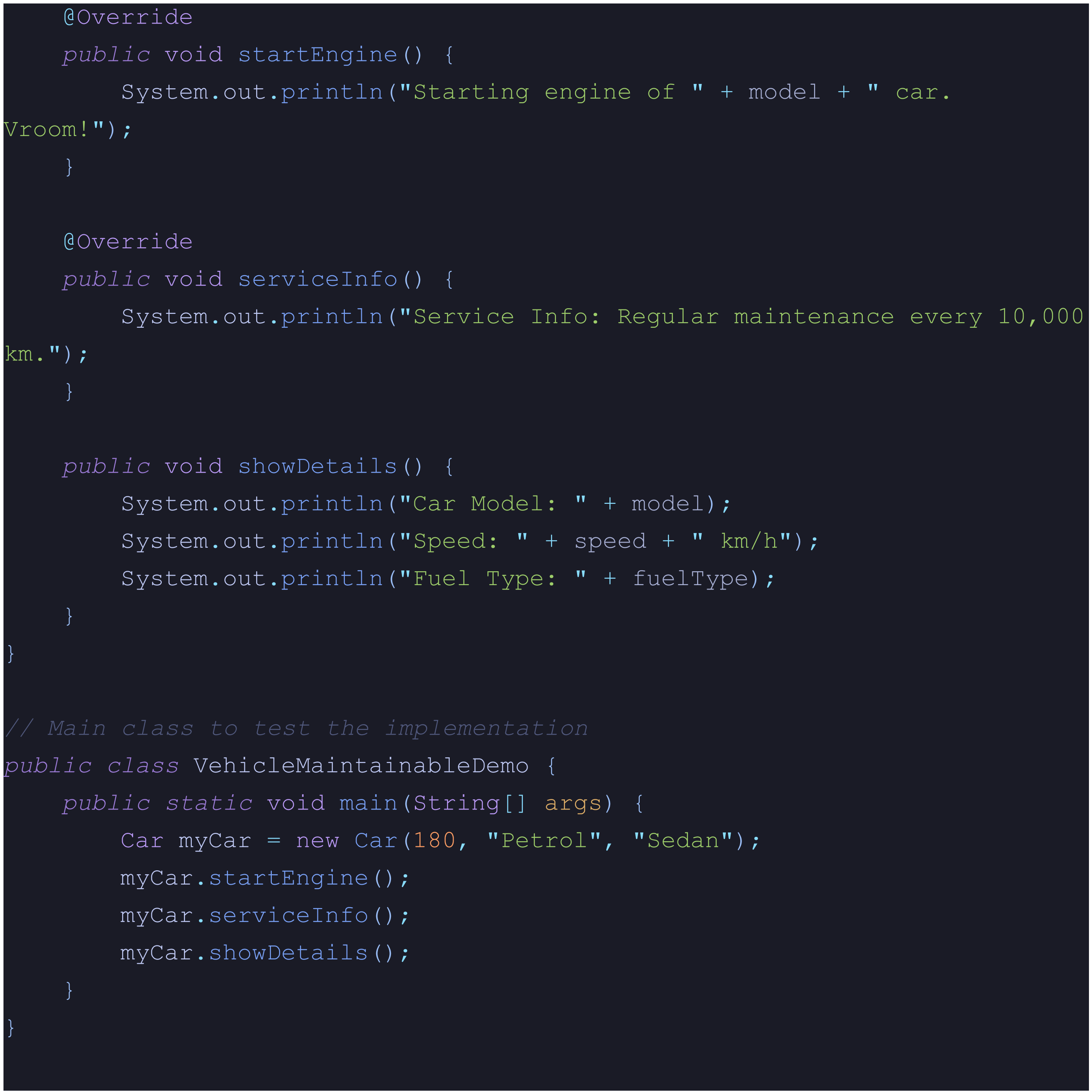
```java
class Circle extends Shape implements Drawable {
    private double radius;

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

    @Override
    public void calculateArea() {
        area = Math.PI * radius * radius;
    }

    @Override
    public void calculatePerimeter() {
        perimeter = 2 * Math.PI * radius;
    }

    @Override
    public void draw() {
        System.out.println("Drawing a circle with radius: " + radius);
    }

    public void showDetails() {
        System.out.println("Circle Radius: " + radius);
        System.out.println("Area: " + area);
        System.out.println("Perimeter: " + perimeter);
    }
}

// Main class to test the implementation
public class ShapeDrawableDemo {
    public static void main(String[] args) {
        Circle myCircle = new Circle(5.0);
        myCircle.calculateArea();
        myCircle.calculatePerimeter();
        myCircle.draw();
        myCircle.showDetails();
    }
}
```



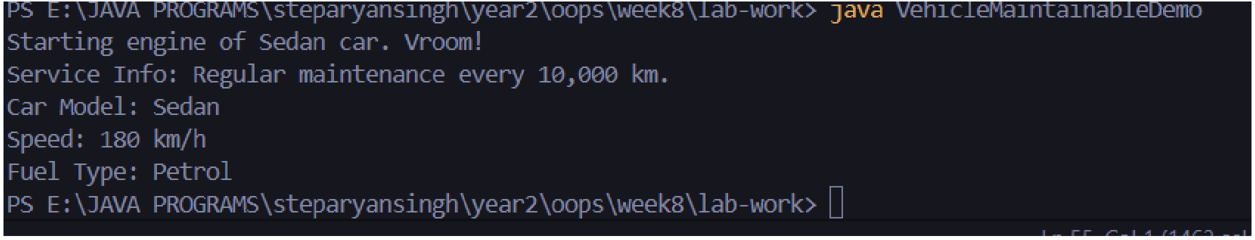
```
PS E:\JAVA PROGRAMS\stepNaryansingh\year2\oops\week8\lab-work>
>> java ShapeDrawableDemo
Drawing a circle with radius: 5.0
Circle Radius: 5.0
Area: 78.53981633974483
Perimeter: 31.41592653589793
```



```java
// Abstract class Vehicle
abstract class Vehicle {
    protected int speed;
    protected String fuelType;

    public Vehicle(int speed, String fuelType) {
        this.speed = speed;
        this.fuelType = fuelType;
    }

    // Abstract method to be implemented by subclasses
    public abstract void startEngine();
}

// Interface Maintainable
interface Maintainable {
    void serviceInfo();
}

// Car class extending Vehicle and implementing Maintainable
class Car extends Vehicle implements Maintainable {
    private String model;

    public Car(int speed, String fuelType, String model) {
        super(speed, fuelType);
        this.model = model;
    }
```

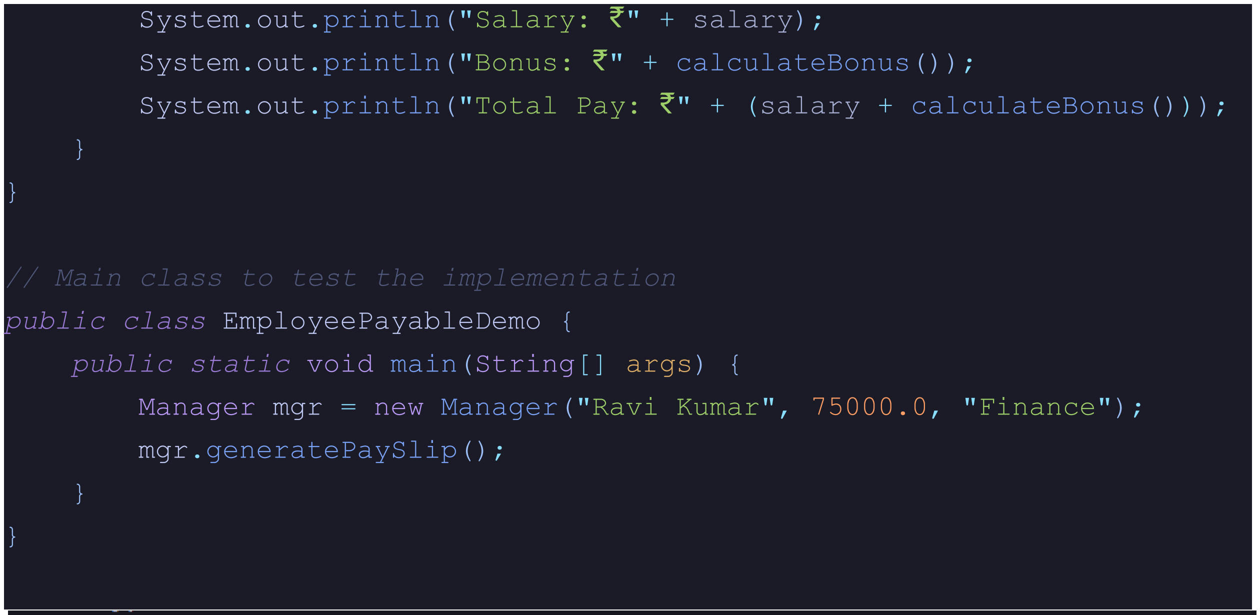


```java
    @Override
    public void startEngine() {
        System.out.println("Starting engine of " + model + " car.
Vroom!");
    }

    @Override
    public void serviceInfo() {
        System.out.println("Service Info: Regular maintenance every 10,000
km.");
    }

    public void showDetails() {
        System.out.println("Car Model: " + model);
        System.out.println("Speed: " + speed + " km/h");
        System.out.println("Fuel Type: " + fuelType);
    }
}

// Main class to test the implementation
public class VehicleMaintainableDemo {
    public static void main(String[] args) {
        Car myCar = new Car(180, "Petrol", "Sedan");
        myCar.startEngine();
        myCar.serviceInfo();
        myCar.showDetails();
    }
}
```

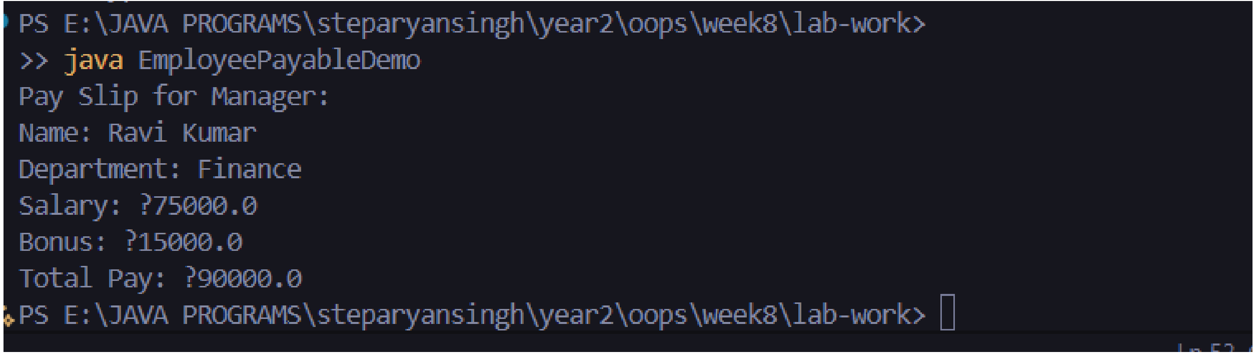


```
PS E:\JAVA PROGRAMS\stepary ansingh\year2\oops\week8\lab-work> java VehicleMaintainableDemo
Starting engine of Sedan car. Vroom!
Service Info: Regular maintenance every 10,000 km.
Car Model: Sedan
Speed: 180 km/h
Fuel Type: Petrol
PS E:\JAVA PROGRAMS\steparyansingh\year2\oops\week8\lab-work> 
```

```java
// Abstract class Employee
abstract class Employee {
    protected String name;
    protected double salary;

    public Employee(String name, double salary) {
        this.name = name;
        this.salary = salary;
    }

    // Abstract method to be implemented by subclasses
    public abstract double calculateBonus();
}

// Interface Payable
interface Payable {
    void generatePaySlip();
}

// Manager class extending Employee and implementing Payable
class Manager extends Employee implements Payable {
    private String department;

    public Manager(String name, double salary, String department) {
        super(name, salary);
        this.department = department;
    }

    @Override
    public double calculateBonus() {
        // Example: 20% of salary as bonus
        return salary * 0.20;
    }

    @Override
    public void generatePaySlip() {
        System.out.println("Pay Slip for Manager:");
        System.out.println("Name: " + name);
        System.out.println("Department: " + department);
```



```java
        System.out.println("Salary: ₹" + salary);
        System.out.println("Bonus: ₹" + calculateBonus());
        System.out.println("Total Pay: ₹" + (salary + calculateBonus()));
    }
}

// Main class to test the implementation
public class EmployeePayableDemo {
    public static void main(String[] args) {
        Manager mgr = new Manager("Ravi Kumar", 75000.0, "Finance");
        mgr.generatePaySlip();
    }
}
```



```
PS E:\JAVA PROGRAMS\stepVaryansingh\year2\oops\week8\lab-work>
>> java EmployeePayableDemo
Pay Slip for Manager:
Name: Ravi Kumar
Department: Finance
Salary: ?75000.0
Bonus: ?15000.0
Total Pay: ?90000.0
PS E:\JAVA PROGRAMS\stepVaryansingh\year2\oops\week8\lab-work>
```