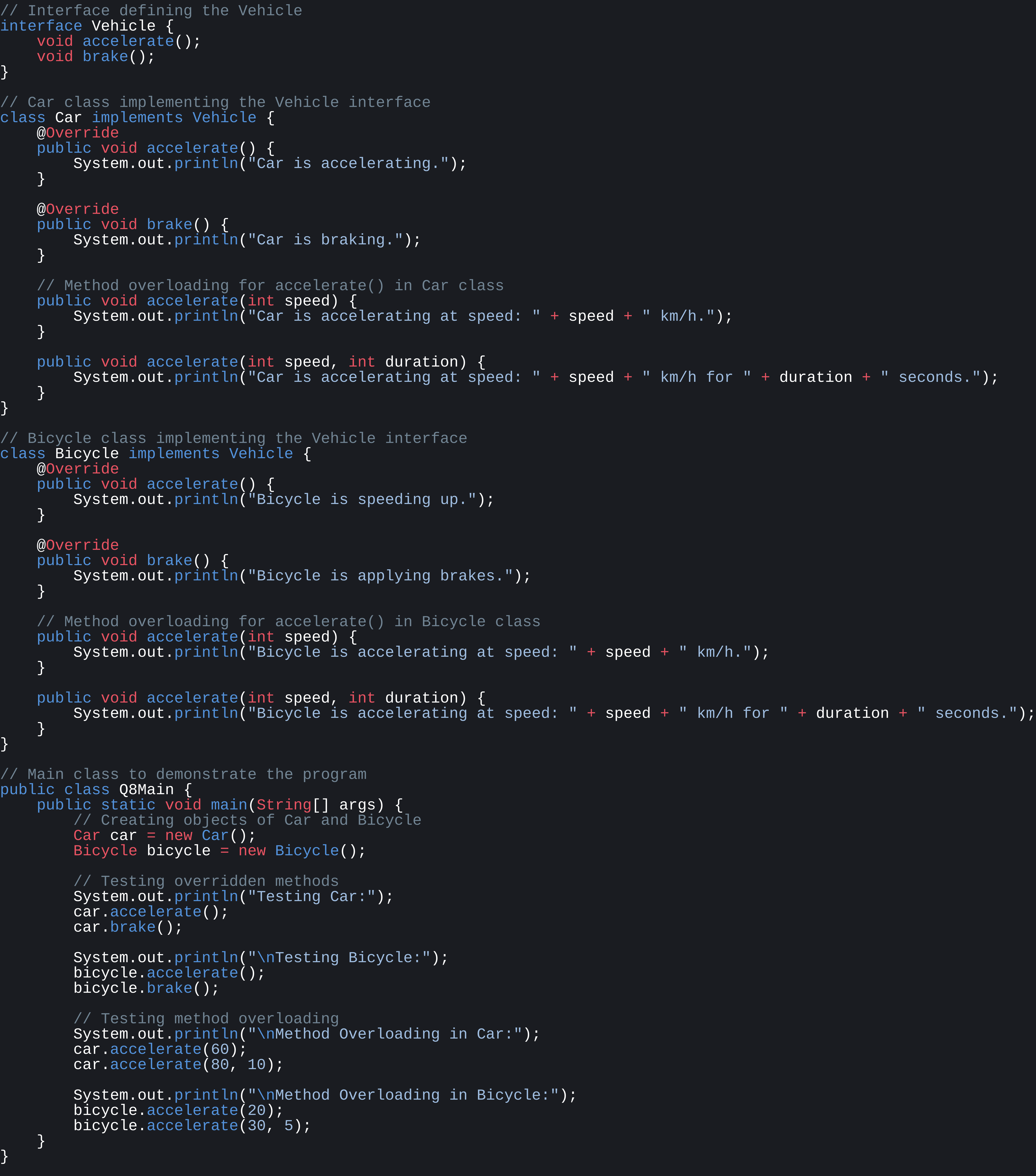

```java
// Interface defining the Vehicle
interface Vehicle {
    void accelerate();
    void brake();
}

// Car class implementing the Vehicle interface
class Car implements Vehicle {
    @Override
    public void accelerate() {
        System.out.println("Car is accelerating.");
    }

    @Override
    public void brake() {
        System.out.println("Car is braking.");
    }

    // Method overloading for accelerate() in Car class
    public void accelerate(int speed) {
        System.out.println("Car is accelerating at speed: " + speed + " km/h.");
    }

    public void accelerate(int speed, int duration) {
        System.out.println("Car is accelerating at speed: " + speed + " km/h for " + duration + " seconds.");
    }
}

// Bicycle class implementing the Vehicle interface
class Bicycle implements Vehicle {
    @Override
    public void accelerate() {
        System.out.println("Bicycle is speeding up.");
    }

    @Override
    public void brake() {
        System.out.println("Bicycle is applying brakes.");
    }

    // Method overloading for accelerate() in Bicycle class
    public void accelerate(int speed) {
        System.out.println("Bicycle is accelerating at speed: " + speed + " km/h.");
    }

    public void accelerate(int speed, int duration) {
        System.out.println("Bicycle is accelerating at speed: " + speed + " km/h for " + duration + " seconds.");
    }
}

// Main class to demonstrate the program
public class Q8Main {
    public static void main(String[] args) {
        // Creating objects of Car and Bicycle
        Car car = new Car();
        Bicycle bicycle = new Bicycle();

        // Testing overridden methods
        System.out.println("Testing Car:");
        car.accelerate();
        car.brake();

        System.out.println("\nTesting Bicycle:");
        bicycle.accelerate();
        bicycle.brake();

        // Testing method overloading
        System.out.println("\nMethod Overloading in Car:");
        car.accelerate(60);
        car.accelerate(80, 10);

        System.out.println("\nMethod Overloading in Bicycle:");
        bicycle.accelerate(20);
        bicycle.accelerate(30, 5);
    }
}
```