

IBM PJP Mentor Session Question to demonstrate is-a, use-a, has-a relationship in Java and JS

JAVA:

```
class Vehicle {  
    protected String brand;  
    protected String model;  
  
    public Vehicle(String brand, String model) {  
        this.brand = brand;  
        this.model = model;  
    }  
  
    public void drive() {  
        System.out.println("Driving " + brand + " " + model);  
    }  
}  
  
// Car class inherits from Vehicle  
class Car extends Vehicle {  
    public Car(String brand, String model) {  
        super(brand, model);  
    }  
  
    public void honk() {  
        System.out.println("Honking the horn of " + brand + " " + model);  
    }  
}  
  
// Bike class inherits from Vehicle  
class Bike extends Vehicle {  
    public Bike(String brand, String model) {  
        super(brand, model);  
    }  
}
```

```
}

    public void ringBell() {
        System.out.println("Ringing the bell of " + brand + " " + model);
    }
}
```

// Customer class has-a Vehicle

```
class Customer {
    private Vehicle vehicle;

    public Customer(Vehicle vehicle) {
        this.vehicle = vehicle;
    }

    public void driveVehicle() {
        vehicle.drive();
    }
}

public class Main {
    public static void main(String[] args) {
        Car myCar = new Car("Toyota", "Camry");
        Bike myBike = new Bike("Giant", "Escape");

        Customer carOwner = new Customer(myCar);
        Customer bikeOwner = new Customer(myBike);

        carOwner.driveVehicle(); // Customer driving the car
        myCar.honk(); // Car honking the horn
        bikeOwner.driveVehicle(); // Customer driving the bike
        myBike.ringBell(); // Bike ringing the bell
    }
}
```

```
}  
}
```

JAVASCRIPT:

```
class Vehicle {  
    constructor(brand, model) {  
        this.brand = brand;  
        this.model = model;  
    }  
  
    drive() {  
        console.log(`Driving ${this.brand} ${this.model}`);  
    }  
}  
  
// Car class inherits from Vehicle  
class Car extends Vehicle {  
    honk() {  
        console.log(`Honking the horn of ${this.brand} ${this.model}`);  
    }  
}  
  
// Bike class inherits from Vehicle  
class Bike extends Vehicle {  
    ringBell() {  
        console.log(`Ringing the bell of ${this.brand} ${this.model}`);  
    }  
}  
  
// Customer class has-a Vehicle
```

```
class Customer {  
    constructor(vehicle) {  
        this.vehicle = vehicle;  
    }  
  
    driveVehicle() {  
        this.vehicle.drive();  
    }  
}
```

// Example usage

```
const myCar = new Car("Toyota", "Camry");  
const myBike = new Bike("Giant", "Escape");
```

```
const carOwner = new Customer(myCar);  
const bikeOwner = new Customer(myBike);
```

```
carOwner.driveVehicle(); // Customer driving the car  
myCar.honk(); // Car honking the horn  
bikeOwner.driveVehicle(); // Customer driving the bike  
myBike.ringBell(); // Bike ringing the bell
```

RIDA UZMAIN SIDDIQUA

10452263