****

****

**信息学院软件工程系**

**《JAVA程序设计》实验报告**

实验六

**姓名：任宇**

**学号：33920212204567**

**学院：信息学院**

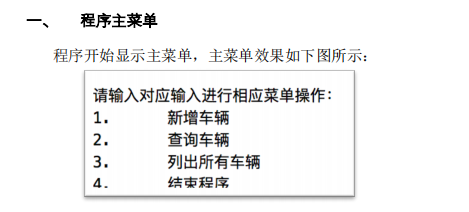
**专业：软件工程**

**完成时间：2024-04-05**

1. **实验目的及要求**

* 熟悉继承
* 尝试多态

1. **实验题目及实现过程**
2. 题目一：
3. 某车队有若干小汽车和卡车，请写一个程序实现新增车辆、查询车辆和列出车辆信息等功能其中，小汽车有商标、颜色、载客量（人）、出厂年、车厢数（2厢或3厢）等属性。卡车有商标、颜色、载重量（吨）、出厂年等属性。



1. 实验环境

集成开发环境：IDEA Community Edition 2022.3.2

JDK版本：JDK17

1. 实现过程

这个车辆管理系统主要涉及以下几个类的设计，以及它们之间的调用关系和用途：

* Vehicle
  + 用途：定义了车辆共有的属性和方法，作为其他具体车辆类型类的基础。
  + 成员：brand（品牌）、color（颜色）、year（出厂年份）
  + 主要方法：

getInformation()：抽象方法，用于获取车辆的详细信息。

* Car
  + 用途：表示小汽车，继承自 Vehicle 类。
  + 成员：增加了 passengerCapacity（载客量）和 carriage（车厢数）。
  + 主要方法：

getInformation ()：重写基类的抽象方法，返回小汽车的具体信息。

* Truck
  + 用途：表示卡车，继承自 Vehicle 类。
  + 成员：增加了 loadCapacity（载重量）。
  + 主要方法：

getInformation ()：重写基类的抽象方法，返回卡车的具体信息。

* VehicleManager
* 用途：管理所有车辆对象，提供车辆的新增、查询和列表显示功能。
* 成员：vehicles（存储 Vehicle 对象的列表）。
* 主要方法：

addVehicle(Vehicle vehicle)：添加新车辆到列表。

listVehicles()：列出所有车辆信息。

searchVehicles(String type, String brand, String color, String year)：根据条件搜索车辆。

* Main
* 用途：实现用户界面，处理用户输入并调用 VehicleManager 的方法执行相应功能。
* 主要方法：

Main(String[] args)：程序的入口方法。

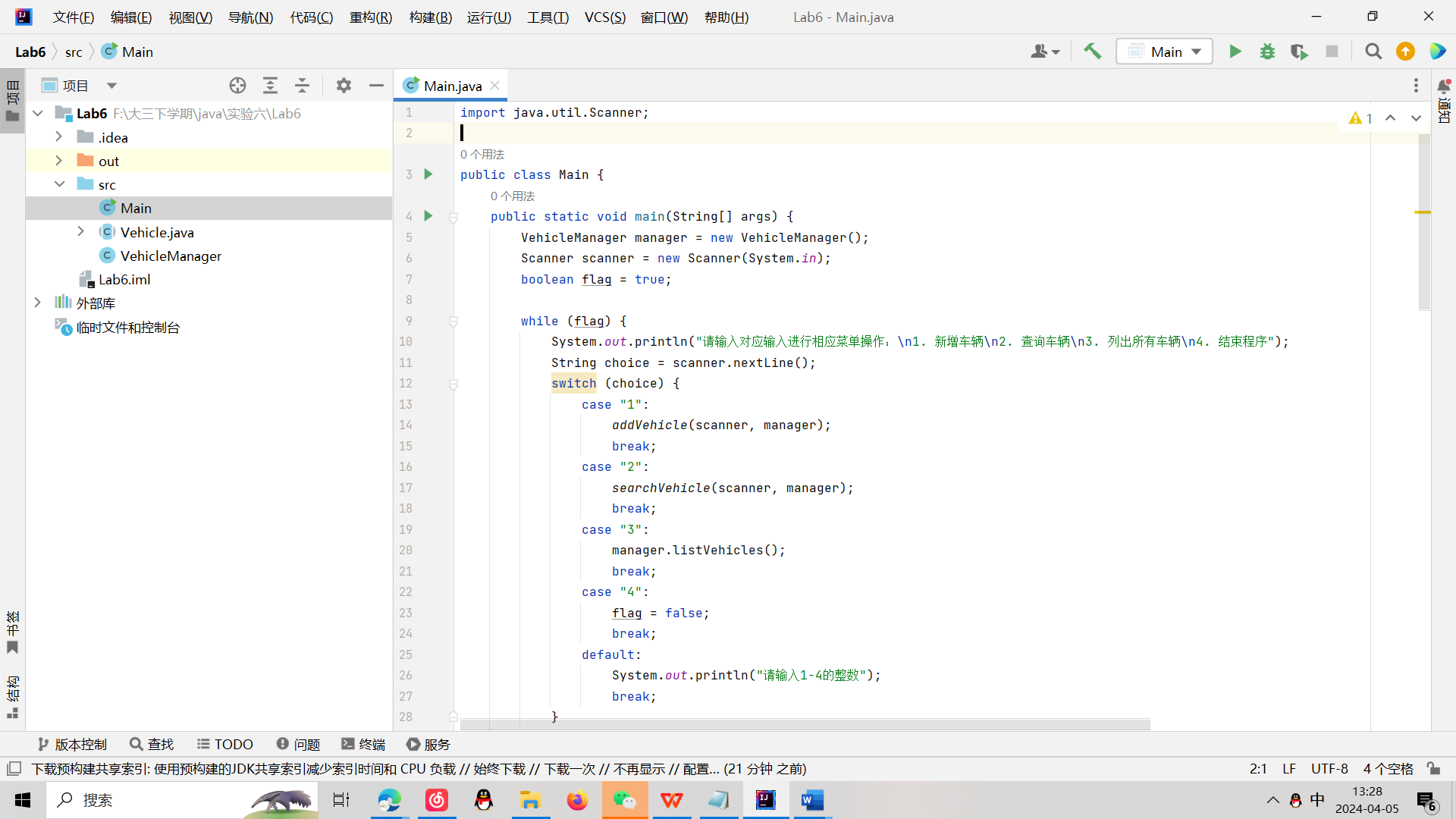
addVehicle(Scanner scanner, VehicleManager manager)：处理添加车辆的用户输入。

searchVehicle(Scanner scanner, VehicleManager manager)：处理搜索车辆的用户输入。

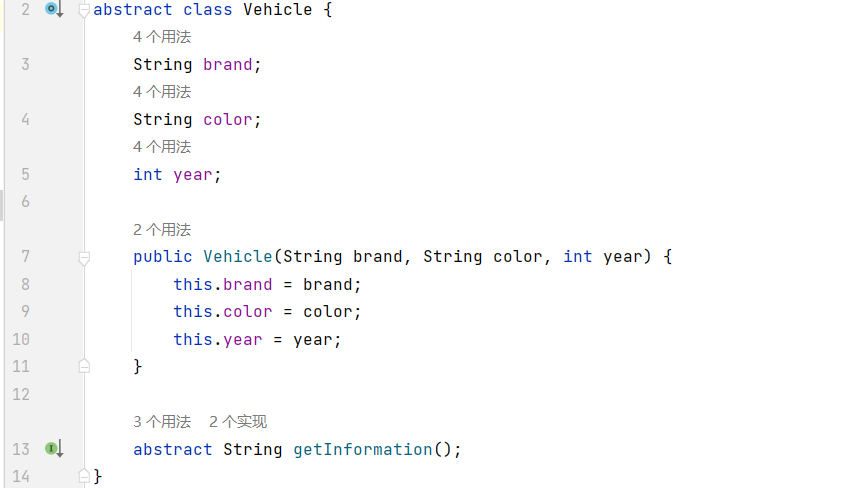
* 调用关系
* 主程序启动：Main 类的 main 方法是程序的起点。它创建一个 VehicleManager 实例来管理车辆，并根据用户的输入调用 addVehicle 或 searchVehicle 方法。
* 添加车辆：当用户选择添加车辆时，Main 类的 addVehicle 方法接收用户输入的车辆信息，创建对应的 Car 或 Truck 实例，并通过 VehicleManager 的 addVehicle 方法将其添加到车辆列表中。
* 搜索车辆：当用户选择搜索车辆时，Main 类的 searchVehicle 方法接收用户输入的搜索条件，并调用 VehicleManager 的 searchVehicles 方法根据条件搜索车辆。
* 列出车辆：当用户选择列出所有车辆时，直接调用 VehicleManager 的 listVehicles 方法显示所有车辆信息。

1. 过程截图

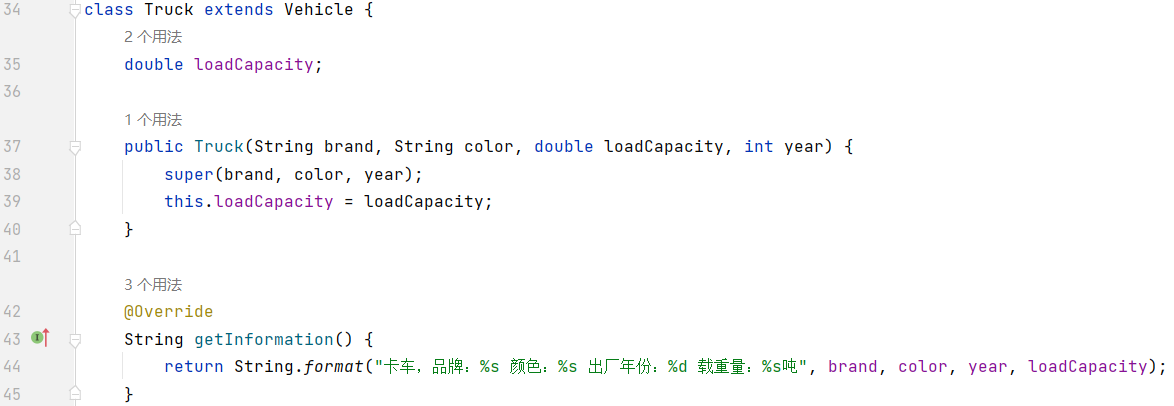
全屏截图：



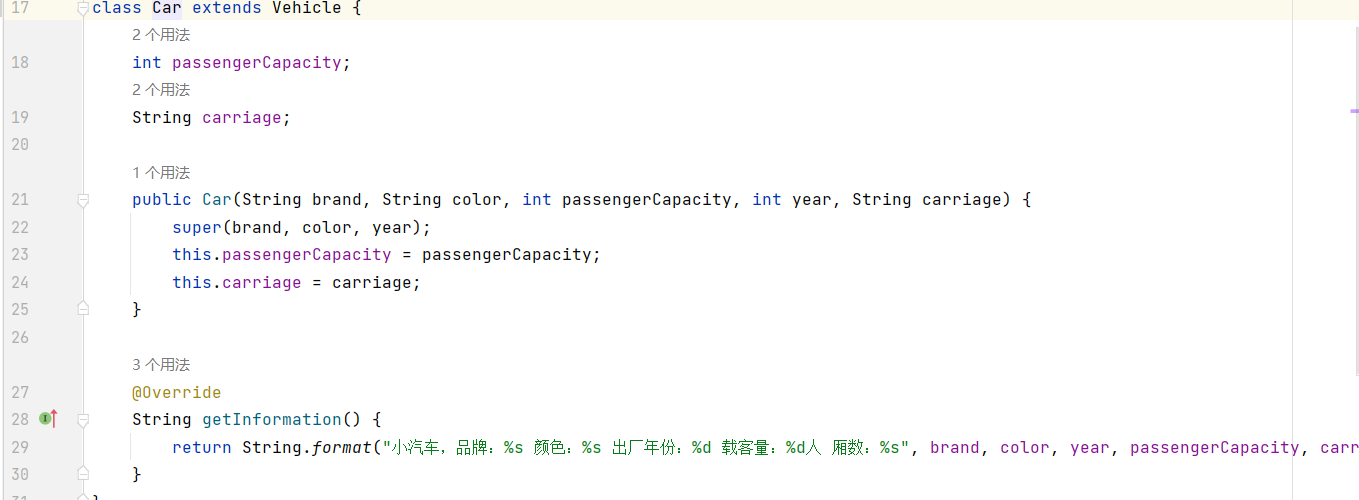
Vehicle类实现：



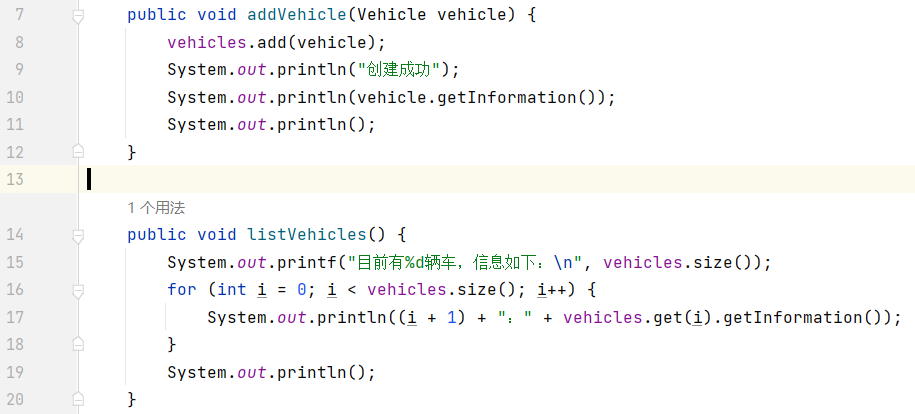
Truck类实现：



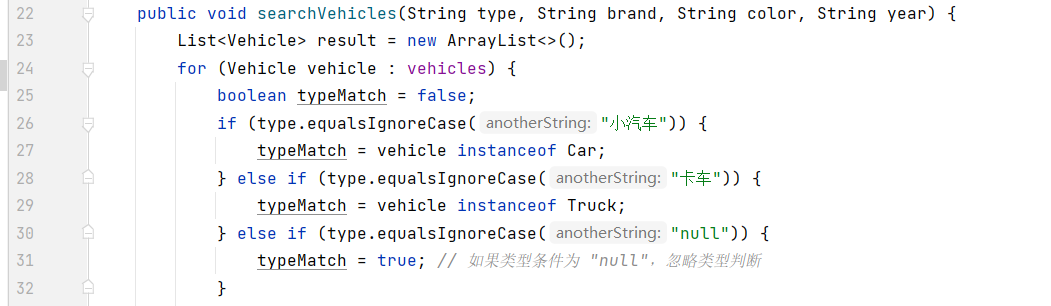
Car类实现：



VehicleManager类实现：



searchVehicles方法实现，方法接收四个字符串参数：车辆类型（type）、品牌（brand）、颜色（color）、以及出厂年份（year），并在内部使用这些条件对车辆列表进行筛选，最后输出搜索到的车辆信息。



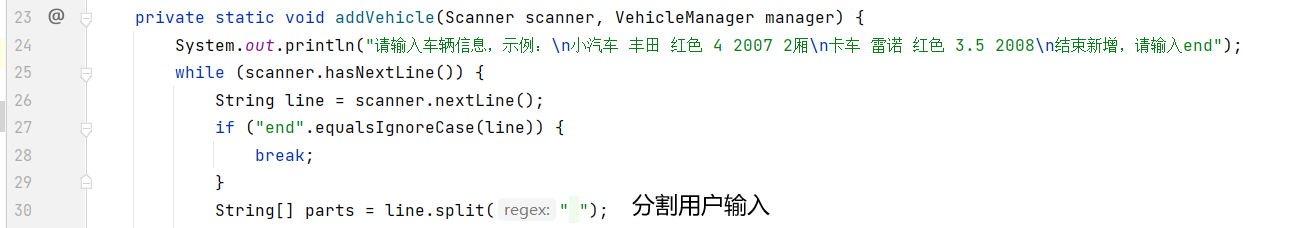


Main类实现：

Main方法，实现程序主菜单以及各个功能的调用。



addVehicle方法，检查并处理用户的输入，用以添加车辆。



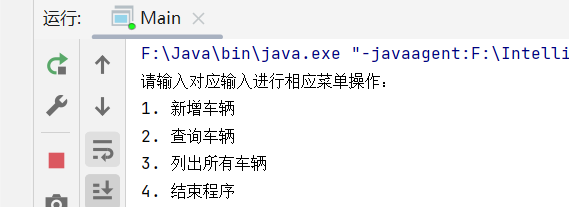


SearchVehicle方法，检查并处理用户的输入，用以搜索指定车辆。

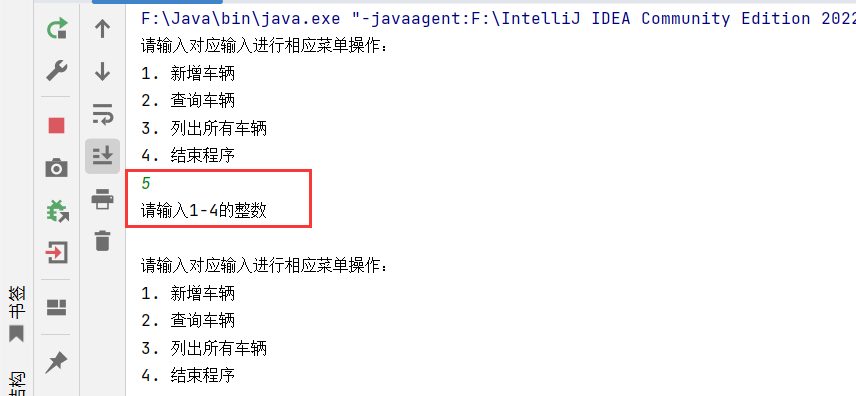


运行程序：

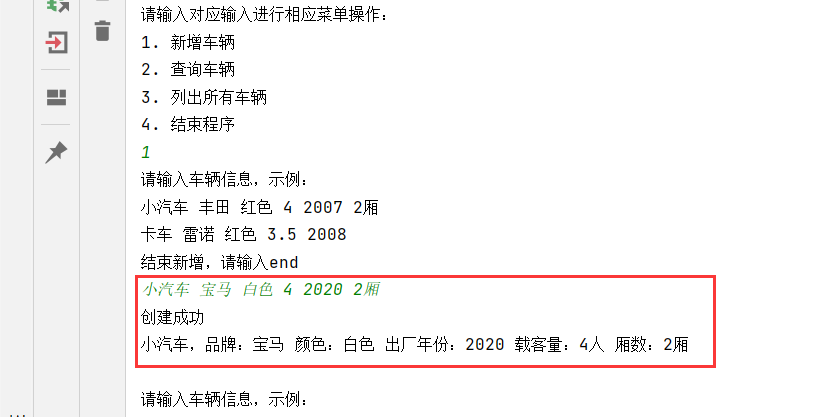
主菜单效果：



主菜单错误检查：

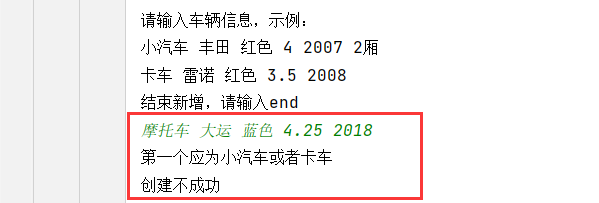


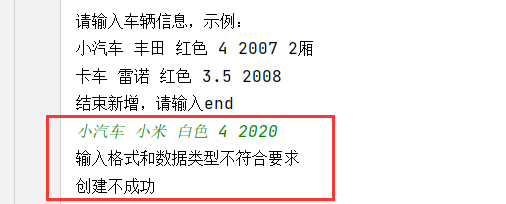
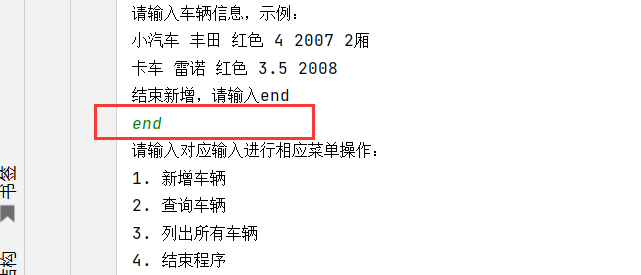
添加车辆：



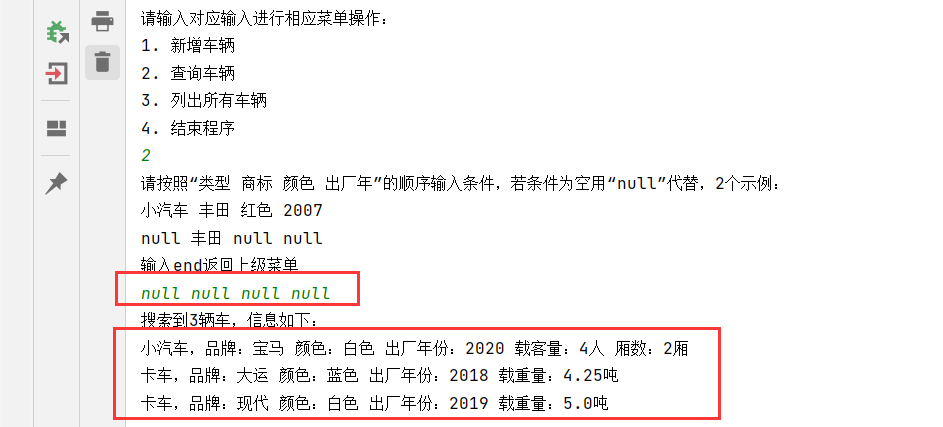


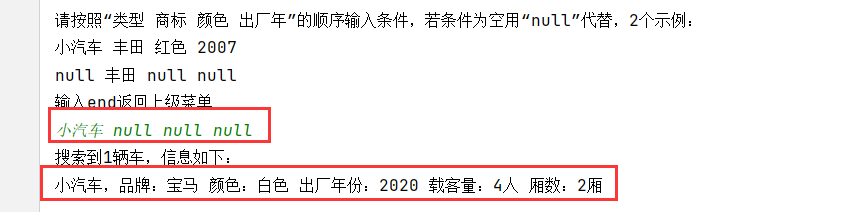
添加车辆错误检查：

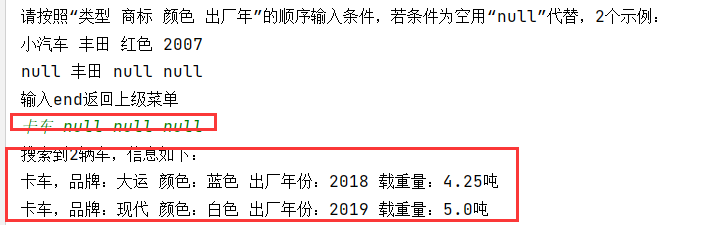


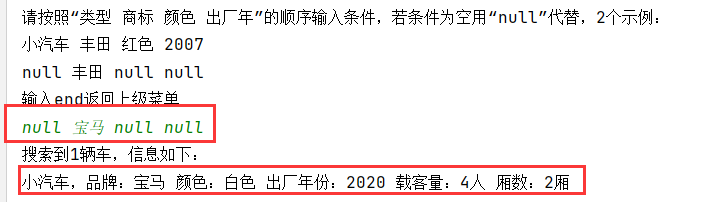
 

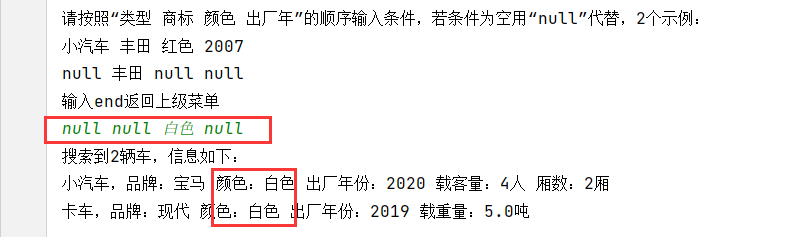
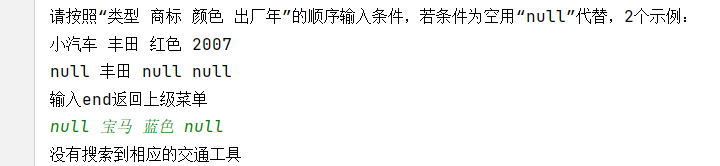
搜索车辆：



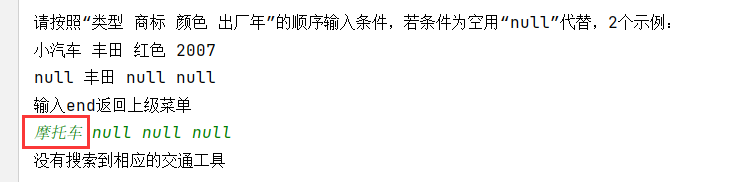


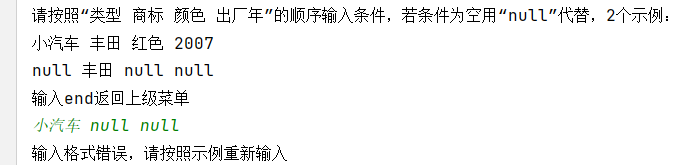




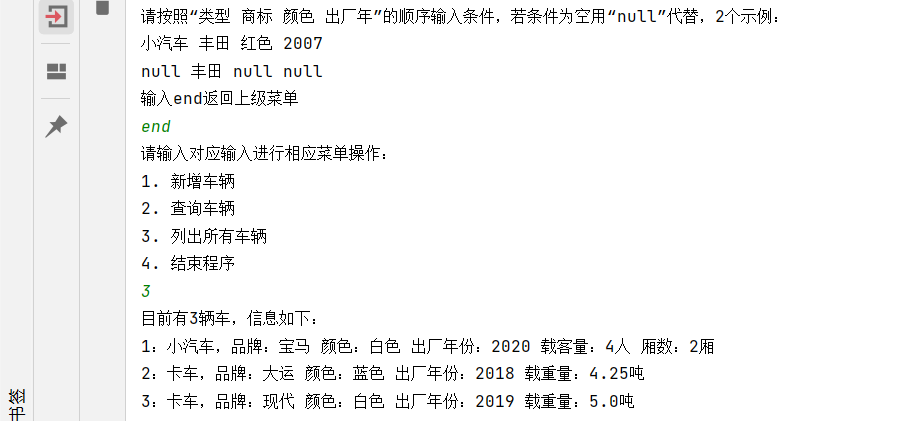
 

搜索车辆错误检查：

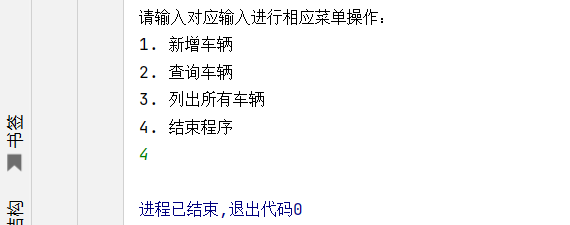




列出所有车辆信息：



退出程序：



**三、实验总结与心得记录**

通过这次设计和实现车辆管理系统的过程，我深刻体会到了面向对象编程中类的继承、封装和多态性的强大用途。通过抽象基类 Vehicle 来定义通用属性和方法，再利用继承让 Car 和 Truck 具体化。实验中，我通过多态性灵活处理不同类型的车辆对象，使得 VehicleManager 类能够统一管理各种车辆，极大地提高了代码的灵活度和可维护性。