【案例6-1】 库存管理系统

### 【案例介绍】

**1.任务描述**

像商城和超市这样的地方，都需要有自己的库房，并且库房商品的库存变化有专人记录，这样才能保证商城和超市正常运转。

本例要求编写一个程序，模拟库存管理系统。该系统主要包括系统首页、商品入库、商品显示和删除商品功能。每个功能的具体要求如下：

1. 系统的首页：用于显示系统所有的操作，并且可以选择使用某一个功能。
2. 商品入库功能：首先提示是否要录入商品，根据用户输入的信息判断是否需要录入商品。如果需要录入商品，则需要用户输入商品的名称、颜色、价格和数量等信息。录入完成后，提示商品录入成功并打印所有商品。如果不需要录入商品，则返回系统首页。
3. 商品显示功能：用户选择商品显示功能后，在控制台打印仓库所有商品信息。
4. 删除商品功能：用户选择删除商品功能后，根据用户输入的商品编号删除商品，并在控制台打印删除后的所有商品。

本案例要求使用Collection集合存储自定义的对象，并用迭代器、增强for循环遍历集合。

**2.运行结果**

商品首页运行结果如图6-1所示。

手机屏幕截图

描述已自动生成

图6-1 商品首页运行结果图

商品入库运行结果如图6-2所示：

手机屏幕截图

描述已自动生成

图6-2 商品入库运行结果图

商品显示运行结果如图6-3所示。

手机屏幕截图

描述已自动生成

图6-3 商品入库运行结果图

商品出库运行结果如图6-4所示：

手机屏幕截图

描述已自动生成

图6-4 商品出库运行结果图

### 【案例任务】

* 学会分析“库存管理系统”任务的实现思路。
* 根据思路独立完成“库存管理系统”任务的源代码编写、编译及运行。
* 掌握Collection以及List集合常用方法的使用。
* 掌握迭代器，增强for循环的使用。

### 【案例思路】

（1）定义仓库产品类，自定义对象定义属性。

（2）在类中定义集合，将要存储的自定义对象作为泛型。

（3）主方法中为集合添加初始化的数据，采用无线循环的方式显示菜单栏。用if...else语句判断用户索要进行的操作，用户可以键盘输入1、2、3来选择对应的操作。如果输入1可以进行商品入库，输入2可以显示库房内所有商品，输入3可以删除进行商品出库，如果不是1、2、3则提示用户操作失败。

（4） 定义商品入库的方法，选择后首先提示用户是否录入商品。输入“no”不录入商品。直接退出并显示所有商品。输入其他任何数据开始录入商品并提示键盘输入什么。录入完毕后退出并显示所有商品。

（5） 定义显示仓库所有商品的方法，用Iterator迭代器的方法循环遍历集合打印，实现商品显示。

（6） 定义商品出库的方法，用集合删除的方法移除要删除的商品。

### 【案例实现】

其中实体类属性如文件6-1所示。

文件6-1 Phone.java

1. public class Phone {
2. private String name;
3. private String color;
4. private double price;
5. private int num;
6. }

以上代码是我们自定义对象实体类所包括的属性，我们在用时需要重写get（）、set（）以及构造方法。

模拟学生管理系统的实现代码，如文件6-2所示。

文件6-2 example1.java

1. package com.itheima;
2. import java.util.ArrayList;
3. import java.util.Collection;
4. import java.util.Iterator;
5. import java.util.List;
6. import java.util.Scanner;
7. public class example1 {
8. static Collection<Phone> c = new ArrayList();
9. public static void main(String[] args) {
10. c.add(new Phone("小米9", "玫瑰金",3999.00,15));
11. c.add(new Phone("小米9", "幻彩紫", 3699, 28));
12. c.add(new Phone("华为P30", "白金色", 5699, 2));
13. c.add(new Phone("华为P30", "绚彩白", 5999, 18));
14. c.add(new Phone("VIVO PLus9", "幻彩紫", 2699, 28));
15. c.add(new Phone("魅族 16th", "紫金黑", 5229, 10));
16. c.add(new Phone("苹果 11", "土豪金", 8999, 51));
17. while(true) {
18. System.out.println("欢迎使用库房管理系统,请选择要进行的操作");
19. System.out.println("1. 商品入库");
20. System.out.println("2. 商品显示");
21. System.out.println("3. 删除商品");
22. Scanner sc = new Scanner(System.in);
23. int s = sc.nextInt();
24. if(s == 1) {
25. addwarehouse();
26. System.out.println("商品入库成功，入库后仓库商品如下：");
27. warehouse();
28. }else if(s==2) {
29. warehouse();
30. }else if(s ==3) {
31. System.out.println("请输入你要删除的商品编号：");
32. int index=sc.nextInt();
33. delwarehouse(index);
34. System.out.println("商品出库成功，出库后仓库商品如下：");
35. warehouse();
36. }else {
37. System.out.println("操作失败！！！");
38. }
39. }
40. }
41. private static void addwarehouse() {
42. while(true) {
43. System.out.println("您是否录入商品？");
44. Scanner w = new Scanner(System.in);
45. String s=w.next();
46. if(!s.equals("no")) {
47. Scanner sc1 = new Scanner(System.in);
48. System.out.println("请输入商品的名称：");
49. String name=sc1.next();
50. System.out.println("请输入商品的颜色：");
51. String color=sc1.next();
52. System.out.println("请输入商品的价格：");
53. int p=sc1.nextInt();
54. System.out.println("请输入商品的数量：");
55. int n=sc1.nextInt();
56. c.add(new Phone(name,color,p,n));
57. break;
58. }else {
59. break;
60. }
61. }
62. }
63. private static void warehouse() {
64. //获取迭代器
65. Iterator it = c.iterator();
66. while(it.hasNext()) {
67. Phone s = (Phone)it.next(); //向下转型
68. System.out.println(s.getName()+ "..." + s.getColor() +
69. "..." + s.getPrice()+ "..." + s.getNum());
70. }
71. }
72. private static void delwarehouse(int index) {
73. c.remove(index);
74. }
75. }

在文件6-2中，第9行定义了一个集合c,泛型是我们的自定义对象Phone。同时定义为静态的。从而实现的代码的复用，减少了代码的冗余。第10~17行代码是为我们定义的集合添加初始化的数据。第18~41行代码采用无线循环的方式显示菜单栏。用if...else语句判断用户索要进行的操作，用户可以键盘输入1、2、3来选择对应的操作。如果输入1可以进行商品入库，调用addwarehouse()方法实现添加商品入库，并且调用warehouse()方法实现入库后商品的显示。输入2可以显示库房内所有商品，调用warehouse()方法即实现商品的显示。输入3可以删除进行商品出库，调用delwarehouse(index)方法实现删除商品，并调用warehouse()方法即实现商品删除后的显示。用如果不是1、2、3则提示用户操作失败。第42~64代码定义商品入库的方法，选择后首先提示用户是否录入商品。输入“no”不录入商品。直接退出并显示所有商品。输入其他任何数据开始录入商品并提示键盘输入什么。录入完毕后退出并显示所有商品。第67~74行代码定义显示仓库所有商品的方法，用Iterator迭代器的方法循环遍历集合打印，实现商品显示。第76~78行代码定义商品出库的方法，用集合删除的方法移除要删除的商品。其中remove()方法是集合移除数据提供的方法。