【案例2-1】 库房出入货物程序设计

### 【案例介绍】

1. **任务描述**

现要对华为和小米两种手机产品进行入库，本案例要求编写一个模拟商品入库的程序，可以在控制台输入入库商品的数量，最后打印出仓库中所有商品详细信息以及所有商品的总库存数和库存商品总金额。

商品信息如下：

* 品牌型号
* 尺寸
* 价格
* 配置
* 库存
* 总价

1. **运行结果**

手机屏幕截图

描述已自动生成

图2-1 运行结果

### 【案例目标】

* 学会分析”库房出入货物”任务实现的逻辑思路。
* 能够独立完成”库房出入货物”程序的源代码编写、编译以及运行。
* 能够在程序中使用算术运算符进行运算操作。
* 能够在程序中使用赋值运算符进行赋值操作。
* 掌握Java中的变量和运算符的知识点。

### 【案例思路】

1. 查看运行结果后，可以将该程序分为3部分实现（商品入库、库存清单、总库存数与库存商品总金额）
2. 商品入库是变化的数据，需要记录商品信息后打印，通过运行结果，我们可以分析出如下属性。

品牌型号：商品的名称，String类型。

尺寸：手机的大小，double类型。

价格：手机的单价，double类型。

配置：手机的内存等配置，String类型。

库存数：此项数据为用户输入的数据，用户输入需要使用Scanner类，以下代码使用户能够从System.in中读取一个数字。

Scanner sc1 = new Scanner(System.in);

int Count=sc1.nextInt();

总价：经过计算后打印，可以设置单独的变量，double类型。

1. 库存清单中又包含了3部分，顶部为固定的数据，直接打印；中部为变化的数据，与商品入库的数据一致，打印出所有商品的详情，底部也为固定样式，直接打印即可。
2. 总库存数与库存商品总金额是统计操作，需经过计算后打印，可以设置两个单独的变量：所有商品的库存总数：int类型。

库存商品总金额：double类型。

### 【案例实现】

文件2-1 access.java

1. package chapter0201;
2. import java.util.Scanner;
3. public class access {
4. public static void main (String[] args) {
5. /\*
6. \*现在有两款手机华为与小米需要做入库处理，我们需要编写一个程序来实现商品的入库，
7. \*入库完成后，打印入库商品的详细信息并计算出入库商品的数量与入库商品总金额。
8. \*/
9. //华为手机
10. String huaweiBrand ="华为";
11. double huaweiSize=5.5;
12. double huaweiPrice=3688.88;
13. String huaweiConfig="8+128g 全面刘海屏";
14. //小米手机
15. String xiaomiBrand ="小米";
16. double xiaomiSize=5.0;
17. double xiaomiPrice=2988.88;
18. String xiaomiConfig="4+64g 全面屏";
19. //华为手机入库
20. System.out.println("品牌型号："+huaweiBrand);
21. System.out.println("尺寸："+huaweiSize);
22. System.out.println("价格："+huaweiPrice);
23. System.out.println("配置："+huaweiConfig);
24. Scanner sc1 = new Scanner(System.in);
25. System.out.println("请输入"+huaweiBrand+"手机的库存");
26. int huanweiCount=sc1.nextInt();
27. double huaweiTotal=huanweiCount\*huaweiPrice;
28. System.out.println("库存"+huaweiBrand+"手机的总金
29. 额:"+huaweiTotal);
30. //小米手机入库
31. System.out.println("品牌型号："+xiaomiBrand);
32. System.out.println("尺寸："+xiaomiSize);
33. System.out.println("价格："+xiaomiPrice);
34. System.out.println("配置："+xiaomiConfig);
35. System.out.println("请输入"+xiaomiBrand+"手机的库存");
36. int xiaomiCount=sc1.nextInt();
37. double xiaomiTotal=xiaomiCount\*xiaomiPrice;
38. System.out.println("库存"+xiaomiBrand+"手机的总金额：
39. "+xiaomiTotal);
40. //库存清单
41. System.out.println("------------库存清单------------");
42. System.out.println("品牌型号 尺寸 价格 配置 库存数
43. 量 总价");
44. System.out.println(huaweiBrand+" "+huaweiSize+"
45. "+huaweiPrice+" "+huaweiConfig+" "+huanweiCount+"
46. "+huaweiTotal);
47. System.out.println(xiaomiBrand+" "+xiaomiSize+"
48. "+xiaomiPrice+" "+xiaomiConfig+" "+xiaomiCount+"
49. "+xiaomiTotal);
50. System.out.println("---------------------------------");
51. int total=huanweiCount+xiaomiCount;
52. double totalMoney=huaweiTotal+xiaomiTotal;
53. //总库存数量与库存总价
54. System.out.println("总库存："+total);
55. System.out.println("库存总价："+totalMoney+"￥");
56. }
57. }

第10~18行代码记录了清单中2种商品的详细信息，包括品牌型号、尺寸、价格、配置和库存数；第20~39行代码是商品入库的代码，先打印商品的详细信息，用户在输入入库的数量，计算出总价，最后打印即可。第41~52行代码用于打印所有商品的信息，需要注意的是，此处商品信息应该与列表顶部的商品信息的顺序一致，否则会造成商品信息显示错乱；第50~55行显示的是清单底部信息，包括总库存数与库存商品的总金额，它们都需要通过计算获得。