【案例2-2】 小明都可以买什么

### 【案例介绍】

**1.任务描述**

编写一个智能购物计算小程序，在一家商店有书本、铅笔、橡皮、可乐、零食五种商品，商品价格如下表所示。

|  |  |
| --- | --- |
| **商品名称** | **价格** |
| 书本 | 12元 |
| 铅笔 | 1元 |
| 橡皮 | 2元 |
| 可乐 | 3元 |
| 零食 | 5元 |

假如你带了20元，且必须购买一本书，剩余的钱还可以购买哪种商品，可以购买几件，购买完后又能剩余多少钱？

**2.运行结果**

任务运行结果如图2-2所示。

手机屏幕截图

描述已自动生成

图2-2 运行结果

### 【案例目标】

* 学会分析"智能购物"程序的实现思路
* 根据思路独立完成”智能购物”的源代码编写、编译及运行。
* 掌握在程序中使用switch条件语句进行运算操作。

### 【案例思路】

1. 从任务描述中可知，要实现此功能，我们需要先定义5种商品，定义五个int值作为这五种商品的价格。
2. 从运行结果可知，我们需要先打印各个商品的价格以及带了多少钱，并选择需要购买商品的序列号。
3. 选择到序列号后，我们需要使用switch条件语句进行判断用户要购买那件商品，并在switch条件语句中，计算可以购买多少其他商品和剩余多少钱。

### 【案例实现】

智能购物的实现代码，如文件2-2所示。

文件2-2 shopping.java

package chapter0202;

import java.util.Scanner;

public class shopping {

public static void main (String[] args) {

/\*

\* 假如你有20元，至少需要购买1本书，剩余的钱还可以购买那些东西。

\*/

int pencil=1; //铅笔价格

int rubber=2; //橡皮价格

int cola=3; //可乐价格

int book=12; //书本价格

int snacks=5; //零食价格

System.out.println("书本的价格为"+book+"元，您总共有20元");

System.out.println("1.铅笔的价格为："+pencil+"元");

System.out.println("2.橡皮的价格为："+rubber+"元");

System.out.println("3.可乐的价格为："+cola+"元");

System.out.println("4.零食的价格为："+snacks+"元");

Scanner sc1 = new Scanner(System.in);

System.out.println("请输入其他需要购买商品的序列号：");

int id=sc1.nextInt();

switch (id){

case 1:

int pencilmoney=20-book;

int pencilsum=pencilmoney/pencil;

int pencilsurplus=pencilmoney%pencil;

System.out.println("购买完书本后还可以购买铅笔"+pencilsum+"

个，还剩"+pencilsurplus+"元");

break;

case 2:

int rubbermoney=20-book;

int rubbersum=rubbermoney/rubber;

int rubbersurplus=rubbermoney%rubber;

System.out.println("购买完书本后还可以购买橡皮"+rubbersum+"

个，还剩"+rubbersurplus+"元");

break;

case 3:

int colamoney=20-book;

int colasum=colamoney/cola;

int colasurplus=colamoney%cola;

System.out.println("购买完书本后还可以购买可乐"+colasum+"个，

还剩"+colasurplus+"元");

break;

case 4:

int snacksmoney=20-book;

int snackssum=snacksmoney/snacks;

int snackssurplus=snacksmoney%snacks;

System.out.println("购买完书本后还可以购买零食"+snackssum+"

个，还剩"+snackssurplus+"元");

break;

default:

System.out.println("您的输入有误。");

break;

}

}

}

第8~12行代码定义了5个商品的价格，第12~17行代码打印了商品的价格，第18~20行代码是让用户输入需要购买的商品序号，第21~53行代码使用了switch条件选择判断用户购买的那中商品，再在switch中计算可以购买多少个以及剩余多少钱。