## 题目

1、有一家超市，出售苹果和草莓。其中苹果 8 元/斤，草莓 13 元/斤。

现在顾客 A 在超市购买了若干斤苹果和草莓，需要计算一共多少钱？

请编写函数，对于 A 购买的水果斤数 (水果斤数为大于等于 0 的整数)，计算并返回所购买商品的总价。

2、超市增加了一种水果芒果，其定价为 20 元/斤。

现在顾客 B 在超市购买了若干斤苹果、 草莓和芒果，需计算一共需要多少钱？

请编写函数，对于 B 购买的水果斤数 (水果斤数为大于等于 0 的整数)，计算并返回所购买商品的总价。

3、超市做促销活动，草莓限时打 8 折。

现在顾客 C 在超市购买了若干斤苹果、 草莓和芒果，需计算一共需要多少钱？

请编写函数，对于 C 购买的水果斤数 (水果斤数为大于等于 0 的整数)，计算并返回所购买商品的总价。

4、促销活动效果明显，超市继续加大促销力度，购物满 100 减 10 块。

现在顾客 D 在超市购买了若干斤苹果、 草莓和芒果，需计算一共需要多少钱？

请编写函数，对于 D 购买的水果斤数 (水果斤数为大于等于 0 的整数)，计算并返回所购买商品的总价。

## 要求

使用 Java 程序编写，IDE 不限，

请使用面向对象的思路进行程序编写。

需在程序中，验证函数计算结果的正确性。

## 提示

顾客购买的水果斤数，可自行确定。无论数值为多少，均需验证程序计算结果的正确性。

可以编写多个函数分别实现，也可以只编写一个函数实现，方式不限。

面试时，请准备电脑及 IDE 环境进行现场演示。

若能将代码提交到 github、gitee、gitlab 等代码托管仓库，提供代码仓库地址更佳。

## 解决方式：

github地址：https://github.com/a8288337/demo.git

（1）设计思路：

需求分析：仔细阅读问题描述，理解顾客购买水果的需求和要求。

数据建模：确定需要的数据模型（Fruit），例如顾客购买的水果种类、数量和价格等。

功能设计：设计计算购买水果总价的函数，根据顾客购买的水果种类和数量，计算总价。

代码实现：根据设计的功能，编写代码实现购买水果总价的计算逻辑。由于每道小题目的需求不一样，我把基础计算方法抽成一个方法，然后根据对应的需求编写对应的方法。

1. 具体实现：

首先，在Application类中，我们定义了一个静态内部类Fruit，用于表示水果的名称、数量、单价和折扣信息。

然后，我们在Application类中定义了两个主要的方法：compute()和specialDiscount()。

compute()方法接收一个可变参数fruits，用于计算购买水果的总价。在该方法中，我们使用一个循环遍历所有的水果，根据水果的单价、数量和折扣计算每种水果的总价，并将其累加到totalPrice变量中。最后，返回总价totalPrice。

specialDiscount()方法接收购买水果的总价totalPrice和满减优惠金额totalDiscount作为参数，用于计算满减后的总价。在该方法中，我们使用一个循环来判断购买总价是否满足满减条件。每当购买总价大于等于100时，我们将满减优惠金额totalDiscount累加10，并将购买总价减去100。最后，返回满减后的总价。

在main()方法中，我们按照题目的要求调用了四个不同的方法：compute1()、compute2()、compute3()和compute4()。这些方法分别计算了不同顾客购买水果的总价，并打印出结果。

每个方法中，我们创建了对应顾客购买的水果对象，并调用compute()方法计算总价。然后，根据题目的要求，打印出顾客购买的水果总价或满减后的总价。

1. 详细代码：

**public** **class** Application {

// flag是否满减优惠劵

**public** **float** compute(Fruit... fruits) {

**float** totalPrice = 0;

**for** (Fruit fruit : fruits) {

totalPrice += fruit.price \* fruit.num \* fruit.discount;

}

**return** totalPrice;

}

**public** **float** specialDiscount(**float** totalPrice, **float** totalDiscount) {

**float** price = totalPrice;

**float** discount = 0;

// 每满100元减10元

**while** (price >= 100) {

discount += totalDiscount;

price -= 100; // 减去满减的金额后，继续判断剩余金额是否满足满减条件

}

**return** totalPrice - discount;

}

**private** **static** **class** Fruit {

String name; // 名称

**int** num; // 数量

**float** price; // 单价

**float** discount; // 折扣

**public** Fruit(String name, **int** num, **float** price, **float** discount) {

**this**.name = name;

**this**.num = num;

**this**.price = price;

**this**.discount = discount;

}

}

**public** **static** **void** main(String[] args) {

/\* (1) 有一家超市，出售苹果和草莓。其中苹果 8 元/斤，草莓 13 元/斤。

现在顾客 A 在超市购买了若干斤苹果和草莓，需要计算一共多少钱？

请编写函数，对于 A 购买的水果斤数 (水果斤数为大于等于 0 的整数)，计算并返回所购买商品的总价。

假设买了8斤苹果，6斤草莓\*/

Application app = **new** Application();

app.*compute1*("A");

app.*compute2*("B");

app.*compute3*("C");

app.*compute4*("D");

}

/\* (1) 有一家超市，出售苹果和草莓。其中苹果 8 元/斤，草莓 13 元/斤。

现在顾客 A 在超市购买了若干斤苹果和草莓，需要计算一共多少钱？

请编写函数，对于 A 购买的水果斤数 (水果斤数为大于等于 0 的整数)，计算并返回所购买商品的总价。

假设买了8斤苹果，6斤草莓\*/

**public** **static** **float** compute1(String customer) {

Application app = **new** Application();

Fruit apple = **new** Fruit("apple", 8, 8, 1);

Fruit strawberry = **new** Fruit("strawberry", 6, 13, 1);

**float** price = app.compute(apple, strawberry);

System.***out***.println(customer + "购买的水果总价：" + price);

**return** price;

}

/\*

\* 2、超市增加了一种水果芒果，其定价为 20 元/斤。 现在顾客 B 在超市购买了若干斤苹果、 草莓和芒果，需计算一共需要多少钱？ 请编写函数，对于 B

\* 购买的水果斤数 (水果斤数为大于等于 0 的整数)，计算并返回所购买商品的总价。 假设买了8斤苹果，6斤草莓，6斤芒果

\*/

**public** **static** **float** compute2(String customer) {

Application app = **new** Application();

Fruit apple = **new** Fruit("apple", 8, 8, 1);

Fruit strawberry = **new** Fruit("strawberry", 6, 13, 1);

Fruit mango = **new** Fruit("mango", 6, 20, 1);

**float** price = app.compute(apple, strawberry, mango);

System.***out***.println(customer + "购买的水果总价：" + price);

**return** price;

}

/\*

\* 3、超市做促销活动，草莓限时打 8 折。 现在顾客 C 在超市购买了若干斤苹果、 草莓和芒果，需计算一共需要多少钱？ 请编写函数，对于 C 购买的水果斤数

\* (水果斤数为大于等于 0 的整数)，计算并返回所购买商品的总价。假设买了8斤苹果，6斤草莓，6斤芒果

\*/

**public** **static** **float** compute3(String customer) {

Application app = **new** Application();

Fruit apple = **new** Fruit("apple", 8, 8, 0.8f);

Fruit strawberry = **new** Fruit("strawberry", 6, 13, 0.8f);

Fruit mango = **new** Fruit("mango", 6, 20, 0.8f);

**float** price = app.compute(apple, strawberry, mango);

System.***out***.println(customer + "购买的水果总价：" + price);

**return** price;

}

/\*

\* 4、促销活动效果明显，超市继续加大促销力度，购物满 100 减 10 块。 现在顾客 D 在超市购买了若干斤苹果、 草莓和芒果，需计算一共需要多少钱？

\* 请编写函数，对于 D 购买的水果斤数 (水果斤数为大于等于 0 的整数)，计算并返回所购买商品的总价。假设买了8斤苹果，6斤草莓，6斤芒果

\*/

**public** **static** **float** compute4(String customer) {

Application app = **new** Application();

Fruit apple = **new** Fruit("apple", 8, 8, 0.8f);

Fruit strawberry = **new** Fruit("strawberry", 6, 13, 0.8f);

Fruit mango = **new** Fruit("mango", 6, 20, 0.8f);

**float** price = app.specialDiscount(app.compute(apple, strawberry, mango), 10);

System.***out***.println(customer + "购买的水果总价（满减后）：" + price);

**return** price;

}

}