

CS1020 Take-home Lab #1

Exercise #1: Redeem Coupon

(http://www.comp.nus.edu.sg/~cs1020/3_ca/labs.html)

Objective

Object-oriented programming.

Task Statement

In general there are two types of coupons: *cash coupon* and *discount coupon*.

Cash coupon has a face value greater than 1, and can be used as cash to buy things. However, the price after discount cannot be less than zero. For example, if you use a \$10.00 coupon to buy a pen worth \$5.80, you would spend \$0 in cash. However, you will not get any change from the shop.

However, there is an “theoretical amount” involved. In the above example, the “theoretical amount” you need to pay is -\$4.20, which means you will “lose” \$4.20 if you use a \$10 coupon to buy the pen. If you use a \$8 coupon instead, the “theoretical amount” will be -\$2.20, which means you “lose” less money. The better choice is hence the \$8 coupon, since you can save the \$10 coupon for something more expensive.

Discount coupon has a discount rate smaller than 1. For example, if the discount rate is 0.2, it means 20% off the price. If you buy a pen worth \$5.80 with this coupon, you would pay \$4.64.

You are to create a Java service class named **Coupon** to represent coupon. The attributes are **name** of a coupon, and **rate** which represents either the face value or discount rate of the coupon. You also need to create a method inside this class to calculate what you need to pay (the “theoretical amount”) when you use a coupon.

You are also to write a client **Redeem** class to read inputs as described below, and compute the best option. If there are two or more options that give the best result, you will pick the first one among them that appears in the input (as illustrated in sample input #3 below).

Input

The first line of the input contains two positive numbers, an amount (of type **double**) and an integer **N** ($1 \leq N \leq 10$) which indicates the number of coupons. In the subsequent **N** lines, each line contains information about a coupon: its name followed by face value or discount rate.

(The above symbol **N** is used to ease explanation. In your program, you should give more descriptive variable names and follow Java naming convention.)

Output

The output contains a single line, stating which is the best coupon to use, and the amount you need to pay, correct to 2 decimal places. (Sometimes, due to rounding differences, your final value may differ from the value in the expected output in the second decimal digit. In this case, as long as your formula is correct, it will be marked as correct.)

Sample Input #1

```
123.35 3
sapphire 20.5
vip 0.3
ruby 30
```

Sample Output #1

```
Best choice: vip
You need to pay $86.34
```

Sample Input #2

```
100.0 3
coupon1 120
coupon2 140
coupon3 110
```

Sample Output #2

```
Best choice: coupon3
You need to pay $0.00
```

Sample Input #3

```
100.0 4
coupon1 0.05
coupon2 120
coupon3 0.1
coupon4 120
```

Sample Output #3

```
Best choice: coupon2
You need to pay $0.00
```

Skeleton Program

The following skeleton is given. Normally, you would have put the **Coupon** class (the server) and **Redeem** class (the client) in separate files, but to reduce the number of files submitted, you are to put them in a single file **Redeem.java**.

```
/*
 * CS1020 (AY2012/3 Sem2)
 * Lab #1 Ex1
 * Author :
 * Matric no.:
 * Description of program:
 */
import java.util.*;

class Coupon {

    // Data attributes
    private String name;
    private double rate;

    // Constructors
    // This default constructor has an empty body
    public Coupon(){
    }
}
```

```

public Coupon(String name, double rate) {
    this.name = name;
    this.rate = rate;
}

// Accessors
public String getName() {
    return this.name;
}

public double getRate() {
    return this.rate;
}

// Write a description for this method
// and complete the method
public double payment(double price) {
}
}

class Redeem {

    public static void main(String[] args) {
        // Declare a Scanner object to read input

        // Declare the necessary variables

        // Read input and process them accordingly

        // Output the result
        // Ensure your output is in the right format
    }
}

```