Clothing Store

A cloth store uses java to maintain information about its cloths. This store has 4 types of cloths (Clothing is abstract):

- Socks
- Shirt
- Pants
- Jacket

Each class has three common properties:

- Name
- Season (a suitable season to use that cloth)
- basePrice

There is also an important and changeable behaviour in this program to calculate each clothes price, considering different discounts (Black Friday, Summer, Winter) such as the following:

Summer discount:

- 40% discount for spring clothes
- 50% discount for summer clothes
- 30% discount for winter clothes

Winter discount:

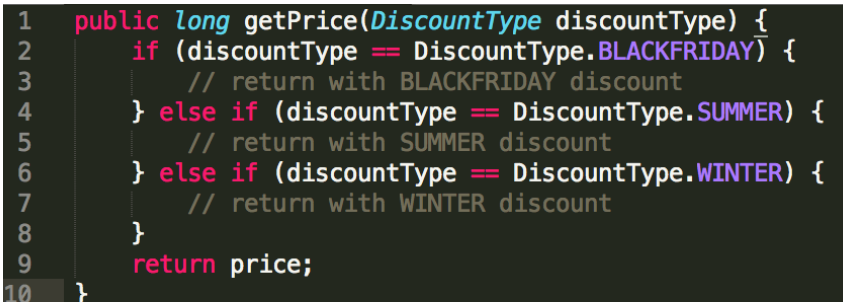
- 40% discount for fall clothes
- 50% discount for winter clothes
- 25% discount for summer clothes
- + 10% discount for all jacket: (ex. one fall jacket has 40% discount first and it gets again extra 10% discount)

Black Friday discount:

- All jackets have 10% discount
- All sucks have 20% discount
- The rest has 25% discount

The programmer had used a method in Clothing class, called getPrice(DiscountType discountType).

In this method, the final price was calculated using the discount type, clothing type and clothing properties:



```java
public long getPrice(DiscountType discountType) {
    if (discountType == DiscountType.BLACKFRIDAY) {
        // return with BLACKFRIDAY discount
    } else if (discountType == DiscountType.SUMMER) {
        // return with SUMMER discount
    } else if (discountType == DiscountType.WINTER) {
        // return with WINTER discount
    }
    return price;
}
```

Now, after some month, the store manager asks the programmer to add some new discounts for different occasions and change some existing discounts' behaviour.

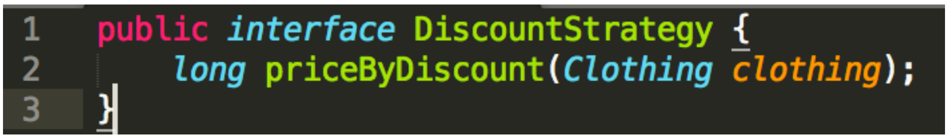
The programmer has found out that there could be a problem in the future:
By adding these changes to getPrice method, or any changes in the future, this method would be bigger and more unclear. So, she has decided to use polymorphism and interface which has learned recently.

To start, she has decided to remove getPrice method, but as she has recently learned these subjects (polymorphism and interface), she needs help to implement her idea, which is your task!

YOUR TASK BEGIN FROM HERE

You should first download InterfaceExercise.

She wants to use one specific strategy class instead of each discount type. To do that, create an interface called DiscountStrategy:

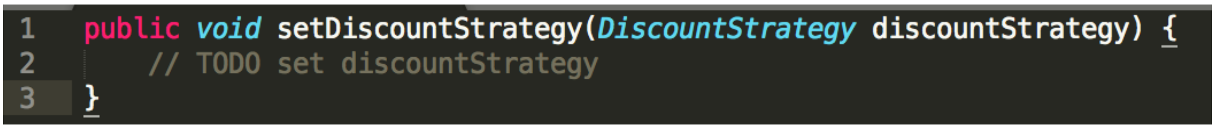


```java
public interface DiscountStrategy {
    long priceByDiscount(Clothing clothing);
}
```

Then, for each discount type create a new class to implement this interface:

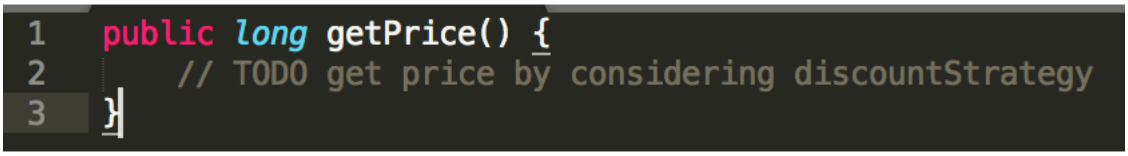
For summer discount: SummerDiscountStrategy class
For winter discount: WinterDiscountStrategy class
For Black Friday discount: BlackFridayDiscountStrategy class

In Clothing class, implement the following method to get and set the current discount type:



```java
public void setDiscountStrategy(DiscountStrategy discountStrategy) {
    // TODO set discountStrategy
}
```

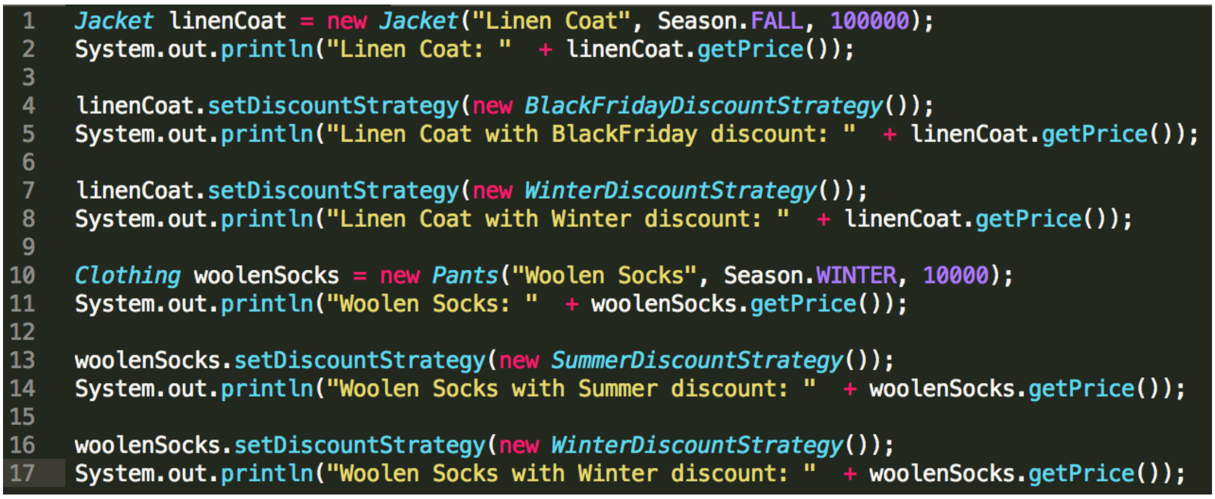
Now in the new getPrice method (without any parameter), calculate the price considering discountStrategy.



```java
public long getPrice() {
    // TODO get price by considering discountStrategy
}
```

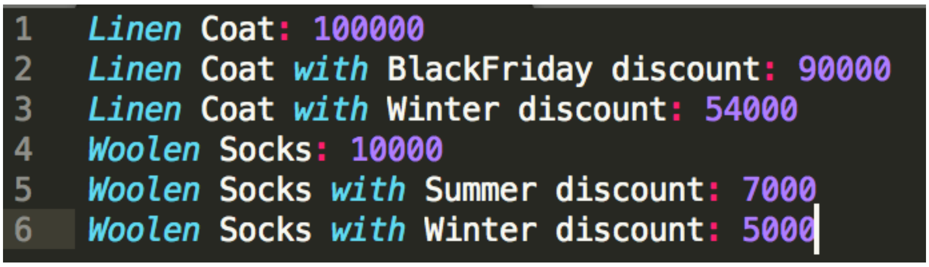
If there is no specified discountStrategy, the basePrice should be returned.

Example:



```java
Jacket linenCoat = new Jacket("Linen Coat", Season.FALL, 100000);
System.out.println("Linen Coat: "  + linenCoat.getPrice());

linenCoat.setDiscountStrategy(new BlackFridayDiscountStrategy());
System.out.println("Linen Coat with BlackFriday discount: "  + linenCoat.getPrice());

linenCoat.setDiscountStrategy(new WinterDiscountStrategy());
System.out.println("Linen Coat with Winter discount: "  + linenCoat.getPrice());

Clothing woolenSocks = new Pants("Woolen Socks", Season.WINTER, 10000);
System.out.println("Woolen Socks: "  + woolenSocks.getPrice());

woolenSocks.setDiscountStrategy(new SummerDiscountStrategy());
System.out.println("Woolen Socks with Summer discount: "  + woolenSocks.getPrice());

woolenSocks.setDiscountStrategy(new WinterDiscountStrategy());
System.out.println("Woolen Socks with Winter discount: "  + woolenSocks.getPrice());
```

The output:



```
Linen Coat: 100000
Linen Coat with BlackFriday discount: 90000
Linen Coat with Winter discount: 54000
Woolen Socks: 10000
Woolen Socks with Summer discount: 7000
Woolen Socks with Winter discount: 5000
```

Good luck!