

0
1

原

设计模式的应用场景(16)--策略模式

2017年08月29日 21:09:41 zhi_fu 阅读数：2212

策略模式

定义：针对一组算法，将每一个算法封装到具有共同接口的独立的类中，使得它们可以互相替换。

优点：替换继承关系，避免使用多重条件转移语句。

缺点：客户端必须知道所有策略类，并自行决定使用哪一种策略类。如果算法较多，则会造成很多的策略类。

使用时机：当系统能在几种算法中快速地切换，或系统中有一些类，它们仅行为不同时，或系统中存在多重条件选择语句时，可以考虑采用策略模式。

小巩需要设计薪资系统，目前各个子公司的算法不一样，怎样才能兼容或切换各个算法呢？

先把薪资三个组成部分的抽象类写出来

```
1 public interface Salary {
2     //计算基本工资
3     public void computerSalary() ;
4 }
5 public interface Insurance{
6     //计算保险
7     public void computerInsurance () ;
8 }
9 public interface Tax{
10    //计算所得税
11    public void computerTax () ;
12 }
13
```


下面实现河北和吉林的算法


```
1 public class HeBeiSalary implements Salary{
2     //计算河北子公司的基本工资
3     public void computerSalary() {
4         System.out.println("采用河北算法计算基本工资");
5     }
6 }
7 public class HeBeiInsurance implements Insurance{
8     //计算河北子公司的保险
9     public void computerInsurance() {
10         System.out.println("采用河北算法计算保险");
11     }
12 }
13 public class HeBeiTax implements Tax{
14     //计算河北子公司的所得税
15     public void computerTax() {
16         System.out.println("采用河北算法计算所得税");
17     }
18 }
19 public class JiLinSalary implements Salary {
20     //计算吉林子公司的基本工资
21     public void computerSalary() {
22         System.out.println("采用吉林算法计算基本工资");
23     }
24 }
25 public class JiLinInsurance implements Insurance {
26     //计算吉林子公司的保险
27     public void computerInsurance() {
28         System.out.println("采用吉林算法计算保险");
29     }
30 }
31 public class JiLinTax implements Tax {
32
```


程序员应该了解的编程语言是~


关闭


```
33 //计算吉林子公司的所得税
34 public void computerTax() {
35     System.out.println("采用吉林算法计算所得税");
36 }
}
```


0

1









模板类，方便策略的切换

```
1 public class SalaryTemplate {
2
3     public void computer () {
4         salary.computerSalary();
5         insurance.computerInsurance();
6         tax.computerTax();
7     }
8
9     public Insurance getInsurance(){ return insurance; }
10
11    public void setInsurance(Insurance insurance){ this.insurance = insurance; }
12
13    public Tax getTax(){ return tax; }
14
15    public void setTax(Tax tax){ this.tax = tax; }
16
17    public Salary getSalary(){ return salary; }
18
19    public void setSalary(Salary salary){ this.salary = salary; }
20
21    private Insurance insurance;
22    private Tax tax;
23    private Salary salary;
24 }
25
```

最后，如果客户想计算吉林的薪资

```
1 public class Client {
2     public static void main(String[] argv) {
3         SalaryTemplate salaryTemplate = new SalaryTemplate();
4         salaryTemplate.setSalary(new JiLinSalary());
5         salaryTemplate.setInsurance(new JiLinInsurance());
6         salaryTemplate.setTax(new JiLinTax());
7         salaryTemplate.computer();
8     }
9 }
```

股票“三不卖七不买”6字黄金口诀，只买并喷股 赢数百万身家！

指南.针 · 顶新



想对作者说点什么

关闭



智雪艳--Kelly： 不同工资的情况分别来写 （11个月前 #1楼）

策略模式和Spring的结合在项目中的应用

阅读数 9849

一、策略模式 策略模式的定义：策略模式是对算法的封装，把使用算法的责任和算法本身分隔开，委派给不同的... 博文 来自： 令仔很忙

设计模式学习之策略模式

阅读数 6万+

写代码时总会出很多的if...else，或者case。如果在一个条件语句中又包含了多个条件语句就会使得代码变得臃肿，维... 博文 来自： 一个码农的博客

模板方法模式和策略模式的应用场景浅析

阅读数 365

标题

博文 来自： laokerr的博客

程序员应该了解的编程语言是~