**Java模板设计模式**

# 一、概要

模板设计模式定义：定义一个操作中的算法骨架，将步骤延迟到子类中。

模板设计模式是一种行为设计模式，一般是准备一个抽象类，将部分逻辑以具体方法或者具体的构造函数实现，然后声明一些抽象方法，这样可以强制子类实现剩余的逻辑。不同的子类以不同的方式实现这些抽象方法，从而对剩余的逻辑有不同的实现。这就是模板设计模式能达成的功能。

适用于一些复杂操作进行步骤分割、抽取公共部分由抽象父类实现、将不同的部分在父类中定义抽象实现、而将具体实现过程由子类完成。对于整体步骤很固定，但是某些部分易变，可以将易变的部分抽取出来，供子类实现。

角色：

抽象类：实现模板方法、定义算法骨架

具体类：实现抽象类中的抽象方法，完成特定的算法

# 二、代码示例

我们举一个比较常见的例子：将一个物品装进冰箱。为了达到这个目的我们一般有如下几步：

a:打开冰箱门

b:将物品装进冰箱

c:关上冰箱门

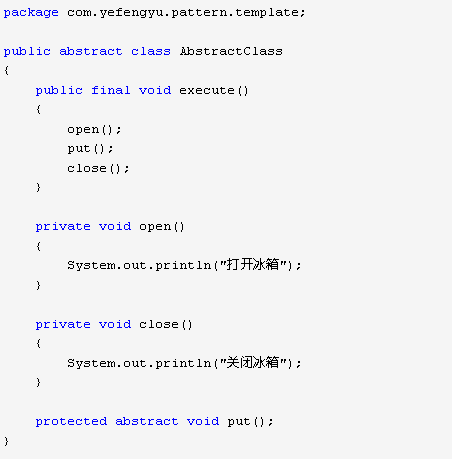
上面的这三步其实就是“将一个物品装进冰箱”这个算法的骨架。在这个算法中，物品这个字眼很重要，它是抽象的，而不是具体的，对于每个不同的物品，装入的时候行为可能不同，这一点非常重要。比如：

a:将一块猪肉放进冰箱--->一块猪肉这么小，直接放进去

b:将一头大象放进冰箱--->一头大象这么大，切碎放进去

上面只是不恰当的举个例子，只是为了说明：针对与不同物品，放入冰箱的动作（行为）不同。

特别注意：上面物品虽是抽象的，但是我最终想表达的是“将物品装进冰箱”这个行为是抽象的。



上面的AbstractClass类中的execute方法就是一个算法骨架，它定义了复杂操作的许多步骤，其中需要注意：

（1）AbstractClass是抽象类，因为子类需要继承某些抽象方法

（2）execute是算法骨架，让外部调用，所以必须是public，但是又不想让子类进行重写，因此使用final关键字，如果重写则导致整个流程混乱。

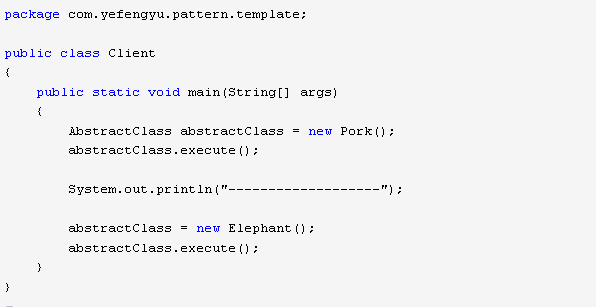
（3）open和close方法是固定的步骤，定义为私有，不让子类修改。

（4）put方法是protected 的，保证子类可以重写，但是其它类无法看到，又是abstract 则子类必须重写。因为父类AbstractClass无法得知具体的物品该如何放入冰箱，只有靠子类去实现了。

下面来实现两个子类。



这两个子类虽然在put中都只打印了一句话，但是我们可以想象这里的操作十分复杂，并且流程大不一样。下面我们来实现客户端代码。



# 三、课后练习

## 1、部门个人评价规则：

**测试人员：**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **姓名** | **部门** | **地区** | **用例数量** | **用例数量等级得分** | **漏检率** | **漏检率等级得分** | **最终考核得分** | **最终考核等级** |

说明：

* 等级得分标准是D得1分，C得2分，B得3分，A得4分。
* 用例数量等级得分根据用例数量数据进行从低到高排序。用例数越大分数越高。
* 漏检率等级得分根据漏检率进行从高到低排序，漏检率越小分数越高。
* 最终考核得分=用例数量等级得分\*0.4+漏检率等级得分\*0.6
* 最终考核等级根据最终考核得分数据进行从低到高排序。
* 其中用例数量等级得分、漏检率等级得分、最终考核等级其分布比例可以动态调整。即参数化。

例如、

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **考核项** | **结果** | **比例** | **数量区间** | **人数** | **考核得分** |
| **用例数量** | D | 56% | ＜= 6 | 24 | 1 |
| C | 16% | 13＜~＜=46 | 7 | 2 |
| B | 16% | 46＜~＜=129 | 7 | 3 |
| A | 12% | 129＜~ | 5 | 4 |

说明：

* 考核分为A、B、C、D四个等级。考核等级比例可以动态调整。
* 用例数量从低到高排序。人数=考核总人数\*考核等级比例，自动计算出对应的区间值。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **考核项** | **结果** | **比例** | **工时区间** | **人数** | **考核得分** |
| **漏检率** | D | 26% | > 23% | 11 | 1 |
| C | 30% | 4%＜~＜=23% | 13 | 2 |
| B | 33% | 2%＜~＜=4% | 15 | 3 |
| A | 12% | <=2% | 5 | 4 |

说明：

* 考核分为A、B、C、D四个等级。考核等级比例可以动态调整。
* 漏检率从高到低排序。人数=考核总人数\*考核等级比例，自动计算出对应的区间值。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **考核项** | **得分区间** | **人数** | **比例** | **考核结果** |
| 最终结果 | 3.5-4 | 4 | 9% | A |
| 2.5-3 | 13 | 30% | B |
| 1.5-2 | 16 | 37% | C |
| 1 | 10 | 23% | D |

说明：

* 考核分为A、B、C、D四个等级。考核等级比例可以动态调整。
* 最终考核得分=用例数量等级得分\*0.4+漏检率等级得分\*0.6
* 最终考核等级根据最终考核得分数据进行从低到高排序。人数=考核总人数\*考核等级比例，自动计算出对应的区间值。

**开发人员：**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **姓名** | **部门** | **地区** | **开发返工比率** | **开发返工比率考核等级得分** | **任务数量** | **按时完成任务数量** | **任务按时完成率** | **任务按时完成率等级得分** | **最终考核得分** | **最终考核等级** |

说明：

* 等级得分标准是D得1分，C得2分，B得3分，A得4分。
* 开发返工比率考核等级得分根据开发返工比率数据进行从低到高排序。越低分数越高。
* 任务按时完成率=按时完成任务数量/任务数量。
* 任务按时完成率考核等级得分根据任务按时完成率数据进行从低到高排序。越高分数越高。
* 最终考核得分=开发返工比率考核等级得分\*0.7+任务按时完成率考核等级得分\*0.3。
* 最终考核等级根据最终考核得分数据进行从低到高排序，最终考核得分，其中开发放工比率，任务按时完成率考核等级得分、最终考核等级其分布比例可动态调整即参数化。

例如、

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **考核项** | **区间** | **人数** | **比例** | **考核结果** | **考核得分** |
| 开发返工比率 | 0%-0.47% | 58 | 29% | A | 4 |
| 0.56%-10% | 72 | 35% | B | 3 |
| 10.69%-35.14% | 65 | 32% | C | 2 |
| 41.21%-83.55% | 8 | 4% | D | 1 |

说明：

* 考核分为A、B、C、D四个等级。考核等级比例可以动态调整。
* 根据开发返工比率数据进行从低到高排序。人数=考核总人数\*考核等级比例，自动计算出对应的区间值。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **考核项** | **区间** | **人数** | **比例** | **考核结果** | **考核得分** |
| 任务按时完成率 | 0.00%-4.76% | 10 | 5% | D | 1 |
| 5.56%-39.53% | 87 | 43% | C | 2 |
| 41.38%-75.00% | 76 | 37% | B | 3 |
| 75.68%-100% | 30 | 15% | A | 4 |

说明：

* 考核分为A、B、C、D四个等级。考核等级比例可以动态调整。
* 根据任务按时完成率数据进行从低到高排序。人数=考核总人数\*考核等级比例，自动计算出对应的区间值。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **考核项** | **区间** | **数量** | **比例** | **考核结果** |
| 最终考核 | 3.1-3.7 | 25 | 12% | A |
| 2.6-3 | 73 | 36% | B |
| 2-2.5 | 91 | 45% | C |
| 1.3-1.9 | 14 | 7% | D |

说明：

* 考核分为A、B、C、D四个等级。考核等级比例可以动态调整。
* 最终考核得分=开发返工比率考核等级得分\*0.7+任务按时完成率考核等级得分\*0.3。
* 根据最终考核得分从高到低排序。人数=考核总人数\*考核等级比例，自动计算出对应的区间值。

## 2、练习要求

计算步骤：

1、获取考核项的A、B、C、D的比例数值。

2、针对这项考核给每个人评分。

3、输出这项考核的统计信息。格式如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **考核项** | **结果** | **比例** | **数量区间** | **人数** | **考核得分** |
| **用例数量** | D | 56% | ＜= 6 | 24 | 1 |
| C | 16% | 13＜~＜=46 | 7 | 2 |
| B | 16% | 46＜~＜=129 | 7 | 3 |
| A | 12% | 129＜~ | 5 | 4 |

备注：上面步骤只是参考，可以按照自己想法设计更合理的计算步骤。