

警告：本文代码未考率 Vector 并发写的问题，而且作者写泛型可能会错

小许:WHAT XX PM,NEW REQUEST

翻译君:XX 产品, 又加需求!

贵公司好, 我们希望能在市民吃饭, 工作时增加一些对物品的需要, 工作时能生产出一些物品.

注 1: 这些物品分种类

注 2: 这些物品可以由政府采购

小许: 什么鬼意思.

BILL(记得吗, 上期 BILL 路过): 就是市民工作时要能搞点东西出来, 吃饭睡觉时要用到这些东西.

小许: 哦

BILL: 我先去休息啦. 你先写着.

小许一段苦思冥想, 写出了头一版代码.

BILL: 小许啊, 我没时间和你整理, 张大胖最近在学习设计模式, 他应该可以帮到你.

---

张 (大胖我就不写了.): 来, 我们分两部分讲. 先说 SKILL 类.

小许: 这怎么了?

张: 你知道吗, 设计模式里有个原则, 叫”一个类, 一个责任”

小许: 可以理解.

张: 那你觉得能否把”资源”放到另一个类中去执行呢?

小许:EMMM, 这题我要看答案.

张: 嘿嘿, 我一开始也没想出来, 你看这段代码 (Decorater.java)(张大胖不会 C++)(我写 JAVA 可能会语法错)

小许: 额,SKILL 的子类套 SKILL, 这不是俄罗斯套娃吗?

张: 是啊, 套一层娃就是多了一个需要, 这就把需要移到别的类里了, 神奇 BUT 强大.

小许: 厉害了.

张: 你先去写, 我喝口水, 一会儿我们继续讲.

---

张: 你说你这 GetR 是不是有”忙等待”的情况啊?

小许: 确实啊.

张:SO?

小许: 那..., 我是不是应该阻塞, 让 NewR 去解除???

张: 嗯, 基本上就是这个思路.

小许: 哦, 那我去写啦.

张: 慢, 加一句, 这个行为也应支持改变.

小许: 嗯, 知道了, 就是把注册的对象从 Need 改为 Function 对吧.

张: 不错啊, 自己连注册都想到了

小许: 嘿嘿嘿

---

BILL: 你这代码 (2/Pattern\_Observer,3/Pattern\_Decorater) 不错啊.

小许: 哪里哪里, 都靠张大胖指点.

BILL: 这里又有两个设计模式, 你查过吗?

小许: 哎呀, 马上去查.

BILL: 不用了你这里套娃的那个用了装饰者模式, 另一个用了观察者模式, 听懂了吧.

小许: 谢谢 BILL 大佬. 嗯?PM 又发需求了?

未完待续.