

【VIP直播课】

# 委派模式及策略模式

Tom





**咕泡学院-Tom**  
**前中电在线技术总监**  
**前超星网架构师**  
**现为咕泡学院联合创始人**

---

10余年Java经验。精通java语言。开发过多套企业UI框架、ORM框架。热衷于分享经验，共同进步。

不只做一个技术者，更要做一个思考者。



## 书法爱好者、绘画爱好者 编程界字写得最好的 书法界编程最牛逼的

自幼开始练习书法。中学期间，曾获市级青少年杯书法竞赛一等奖，获校园杯美术竞赛工笔画一等奖，获校园征文比赛二等奖。大学担任学生会宣传部长，负责校园黑板报、校园刊物的编辑排版设计。参加工作后，担任过家具建模、平面设计等工作。

亲自设计咕泡学院Logo。

- 1、掌握委派模式，精简程序逻辑，提升代码的可读性。
- 2、通过学习策略模式来消除程序中大量的if...else...和switch语句。
- 3、深刻理解策略模式的应用场景，提高算法的保密性和安全性。

- 1、希望通过对委派模式的学习，让自己写出更加优雅的代码的人群。
- 2、希望通过对策略模式的学习，来消除程序中大量的冗余代码和多重条件转移语句的人群。

# 1

## 委派模式详解



**委派模式（Delegate Pattern）的基本作用就是负责任务的调度和分配任务，跟代理模式很像，可以看做是一种特殊情况下的静态代理的全权代理，但是代理模式注重过程，而委派模式注重结果。**

**不属于GOF 23种设计模式之一。**

**属于行为型模式。**

**Delegate 结尾的一般都是委派， Dispatcher**





## 策略模式详解





**策略模式 (Strategy attern)** 是指定义了算法家族、分别封装起来，让它们之间可以互相替换，此模式让算法的变化不会影响到使用算法的用户。

**可以避免多重分支的if...else...和switch语句**



- 1、假如系统中有很多类，而他们的区别仅仅在于他们的行为不同。
- 2、一个系统需要动态地在几种算法中选择一种。



- 1、策略模式符合开闭原则。
- 2、避免使用多重条件转移语句，如if...else...语句、switch语句
- 3、使用策略模式可以提高算法的保密性和安全性。



- 1、客户端必须知道所有的策略，并且自行决定使用哪一个策略类。
- 2、代码中会产生非常多策略类，增加维护难度。



# 谢谢观看

Tom



Tom老师QQ号: 441221062

