

qq 交流群：894294771

第二十五章 行为型模式

行为型模式用于描述程序在运行时复杂的流程控制,即描述多个类或对象之间怎样相互协作共同完成单个对象都无法单独完成的任务,它涉及算法与对象间职责的分配。

行为型模式分为类行为模式和对象行为模式,前者采用继承机制来在类间分派行为,后者采用组合或聚合在对象间分配行为。由于组合关系或聚合关系比继承关系耦合度低,满足“合成复用原则”,所以对象行为模式比类行为模式具有更大的灵活性。

行为型模式是 GoF 设计模式中最为庞大的一类,它包含以下 11 种模式:

1) 模板方法 (Template Method) 模式: 定义一个操作中的算法骨架,将算法的一些步骤延迟到子类中,使得子类在不改变该算法结构的情况下重定义该算法的某些特定步骤。

2) 策略 (Strategy) 模式: 定义了一系列算法,并将每个算法封装起来,使它们可以相互替换,且算法的改变不会影响使用算法的客户。

3) 命令 (Command) 模式: 将一个请求封装为一个对象,使发出请求的责任和执行请求的责任分割开。

4) 职责链 (Chain of Responsibility) 模式: 把请求从链中的一个对象传到下一个对象,直到请求被响应为止。通过这种方式去除对象之间的耦合。

5) 状态 (State) 模式: 允许一个对象在其内部状态发生改变时改变其行为能力。

6) 观察者 (Observer) 模式: 多个对象间存在一对多关系,当一个对象发生改变时,把这种改变通知给其他多个对象,从而影响其他对象的行为。

7) 中介者 (Mediator) 模式: 定义一个中介对象来简化原有对象之间的交互关系,降低系统中对象间的耦合度,使原有对象之间不必相互了解。

8) 迭代器 (Iterator) 模式: 提供一种方法来顺序访问聚合对象中的一系列数据,而不暴露聚合对象的内部表示。

9) 访问者 (Visitor) 模式: 在不改变集合元素的前提下, 为一个集合中的每个元素提供多种访问方式, 即每个元素有多个访问者对象访问。

10) 备忘录 (Memento) 模式: 在不破坏封装性的前提下, 获取并保存一个对象的内部状态, 以便以后恢复它。

11) 解释器 (Interpreter) 模式: 提供如何定义语言的文法, 以及对语言句子的解释方法, 即解释器。

以上 11 种行为型模式, 除了**模板方法模式**和**解释器模式**是类行为型模式, 其他的全部属于对象行为型模式, 下面我们将详细介绍它们的特点、结构与应用。