# 设计模式

## 原则

开闭原则,对扩展开放,对修改封闭

设计模式共有 23 种, 其中,创建型模式 5种 ,结构性模式 7种,行为模式 11 种

创建型模式 包括 简单工厂,工厂方法,抽象工厂,建造者,单例

结构性包括 桥接,享元模式,适配器模式,装饰模式外观模式,代理模式,组织模式

## 创建型模式(5种) 使用场景 创建类

### 简单工厂模式

通过参数创建类

### 工厂方法模式

先创建工厂,然后创建类

### 抽象工厂模式

创建抽象工厂,抽象工厂创建简单工厂,简单工厂创建类

### 建造者模式

不需要全部初始化,提供默认参数

### 单利模式

只有一个实例

## 结构性模式(7种) 使用场景 定义类

### 适配器模式

两个系统对接,将调用双方代码提供出一个类

### 桥接模式

将逻辑定义和实现分离开,多用于架构设计,spring

### 装饰模式

通过继承扩充父类方法

### 外观模式

将内部代码逻辑进行整合,向上提高一个逻辑层次

### 享元模式

将一列类具有不同特征的相同对象保存在一个数组种,减少每次使用前都创建对象 可以和单利模式组合使用

### 代理模式

通过代理对象,增加切面方法

## 行为模式 (11 种) 使用场景 两个类之间的交互

### 命令模式

将参数封装,告诉让谁做什么事(spring mvc)

### 中介者模式

通过中介者将 A和B联系起来,操作A的同时,实际上是通过中介者操作B

### 观察者模式

通过观察者数组或被观察者引用,更新状态 推模式和拉模式

### 状态模式

将系统的每一种状态创建一系列的状态对象 openclass closedclass

### 策略模式

利用多态的特性,创建合适的对象