

设计模式

题型

简答、应用

题目

1.设计一个新模式

Get your ideas down **on the** paper **in** a way others can understand.
结合一个已有的模式定义一个新的模式。

2.Thinking in pattern

kiss(Keep it simple)
对不需要的部分就不使用

复习

工厂

1. 工厂对产品有什么不同的要求？
2. 设计原则 对开闭原则的支持程度（哪些修改支持，哪些不支持）
3. 设计细节
4. 工厂之间的关系
5. 工厂方法的优缺点（见PPT）

简单工厂对开闭支持最差，工厂方法符合“开闭原则”
工厂方法，工厂弗雷负责定义产品对象的公共接口，工厂子类负责生成具体的产品对象，将产品的实例化操作延迟到子类

工厂类的大例证在于延迟到了类。

简单工厂简化了产品的生产过程，工厂方法通过继承来实现灵活性。

三种工厂方法的优缺点

抽象工厂的退化

当抽象工厂模式中每一个具体工厂类只创建一个产品对象，也就是只存在一个产品等级结构时，抽象工厂模式退化成工厂方法模式；当工厂方法模式中抽象工厂与具体工厂合并，提供一个统一的工厂来创建产品对象，并将创建对象的工厂方法设计为静态方法时，工厂方法模式退化成简单工厂模式。

单例模式

主要掌握代码实现，有三个注意点

单例类的构造函数为私有；
提供一个自身的静态私有成员变量；
提供一个公有的静态工厂方法。

优缺点

适配器模式

（画类图注意实线虚线） 适配器的扩展，如何对适配器模版进行双向

双向适配器P32

适配器别名Wrapper，装饰器别名也为Wrapper

适配器，同时实现Target接口还有Adaptee的方法

对象适配器 关联关系
类适配器

优缺点

装饰模式

符合“开闭原则”

缺点：

产生小对象，但是类型数不会很多
比继承更加易于出错，排错困难，对于多次装饰的对象，题哦啊是寻找错误可能需要逐级排查，较为繁琐

适用环境：

动态增加功能，动态撤销功能

需要自己会写装饰模式

判断Java IO中 inputStream等，谁是最基本的，不能说错

组合模式

肯定要考，而且和别的模式一起考

树形结构——层次关系

访问上的使用

有区别的对待容器对象和叶子对象

组合模式递归访问出现在什么地方

每个component都有operation

模式透明性（透明性和安全性的考量）

客户代码希望变化是透明的

客户代码希望变化是透明的

更注重容器内组合结构的时候，可以牺牲透明，实现安全组合模式

享元模式Flyweight Pool

工厂模式和享元模式可以结合

享元的内部状态和外部状态

享元工厂