

设计模式简单来说,就是写代码时的套路。

单例(Singleton)设计模式 —— 饿汉式:

这个设计模式用于整个程序只需要一个实例化对象的情况。

```
class Singleton {  
    private Singleton c;  
    // 私有构造函数  
    private static Singleton a = new Singleton();  
    public static Singleton getA() {  
        return a;  
    }  
}
```

还有一个饿汉式设计模式,与饿汉式的区别是,饿汉式中new的对象在类加载时就存在,懒汉式则必须调用该方法才new对象。

模板方法设计模式(Template Method)

抽象类体现的就是一种模板模式的设计,抽象类作为整个类的通用模板,子类在抽象类的基础上进行扩展、改造,但其保留了抽象类的行为模式。

解决的问题:

- 当功能内部一部分实现是确定的,一部分是不确定的,可以将不确定的部分交给子类实现。

工厂方法(Factory Method):