**spring aop**

**1、本章面试题**

           什么是AOP，说说项目中如何使用

**2、知识点**

**2.1、课程回顾**

       什么是IOC DI 三种方式 （设值注入 构造和注解）

**2.2、本章重点**

         AOP概念

AOP 专门术语

AOP配置和注解两种实现方式

**3、具体内容**

**3.1 AOP概念**

AOP（Aspect-Oriented Programming，面向切面编程），简单地说，就是将那些与业务无关，却为业务模块所共同调用的逻辑或业务封装起来，便于减少系统的重复代码，降低模块间的耦合度，并有利于未来的可操作性和可维护性。

AOP的核心思想就是“将应用程序中的业务逻辑同对其提供支持的通用服务进行分离"

例如：将日志记录，性能统计，安全控制，事务处理，异常处理等代码从业务逻辑代码中划分出来，通过对这些行为的分离，我们希望可以将它们独立到非指导业务逻辑的方法中，进而改变这些行为的时候不影响业务逻辑的代码。

**3.2 AOP入门示例**

需求：在服务层用AOP给所有方法添加统一日志服务

在ioc基础上添加依赖

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-aop</artifactId>

<version>${spring.version}</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.aspectj/aspectjweaver -->

<dependency>

<groupId>org.aspectj</groupId>

<artifactId>aspectjweaver</artifactId>

<version>1.9.4</version>

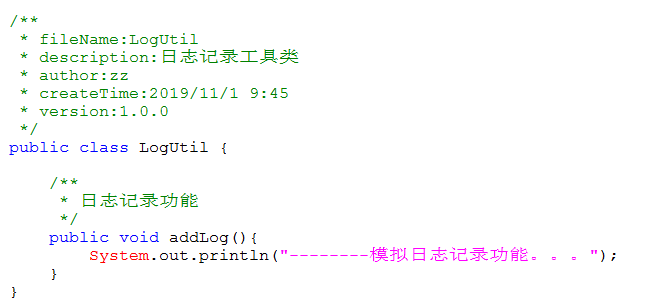
</dependency>

1，编写服务接口和服务类

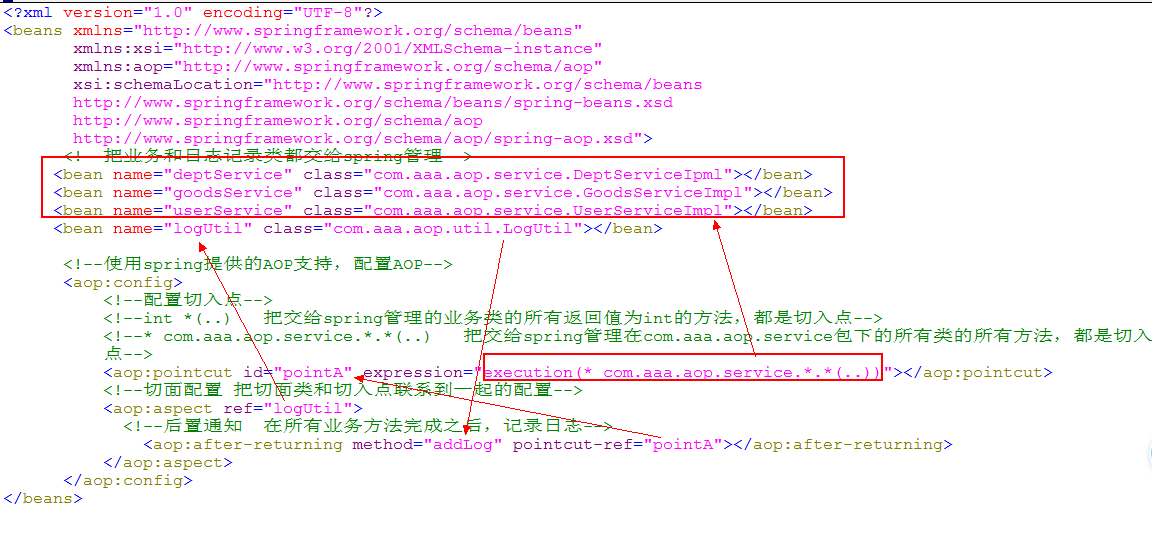




2，编写日志记录工具类



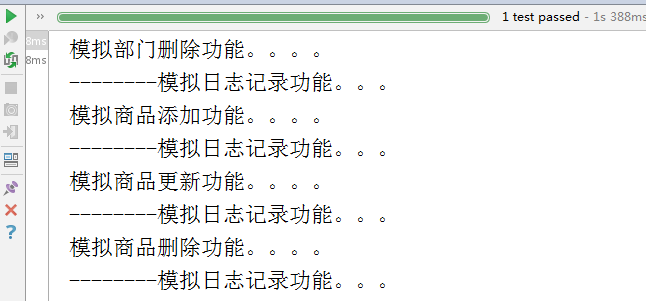
3，aop配置：



4，编写测试类：



测试结果



**3.3 aop术语简介**

 **切面**（Aspect）

如：将日志记录，性能统计，安全控制，事务处理，异常处理

 **通知**(Adivce)

通知是切面的具体实现，是放置“切面代码”的类。

 **切入点**（Pointcut）

多个连接点组成一个切入点，可以使用切入点表达式来表示。

  **切入点表达式：**

任意公共方法的执行：

execution（public \* \*（..））

任何一个名字以“set”开始的方法的执行：

execution（\* set\*（..））

AccountService接口定义的任意方法的执行：

execution（\* com.xyz.service.AccountService.\*（..））

在service包中定义的任意方法的执行：

execution（\* com.xyz.service.\*.\*（..））

在service包或其子包中定义的任意方法的执行：

execution（\* com.xyz.service..\*.\*（..））

**连接点**（Join point）

程序执行过程中某个特定的点，连接点总是代表某个方法的执行，业务接口的方法均为连接点。

 **织入**（Weaving）

织入是指将切面代码插入到目标对象的过程（或者将切入点和通知结合的过程称为织入）。

 **目标对象**（Target）

包含主业务逻辑的类的对象，包含一个连接点的对象，即被拦截的对象。

**3.4 通知类型**

 Before：在目标方法被调用之前调用（前置通知）

 AfterReturning：在目标方法被调用之后调用（后置通知）

 Around：拦截对目标对象方法的调用（环绕通知）

 After：无论目标方法是否正常运行，都一定要执行的通知（最终通知）

 AfterThrowing：当目标方法抛出异常时调用（异常通知）

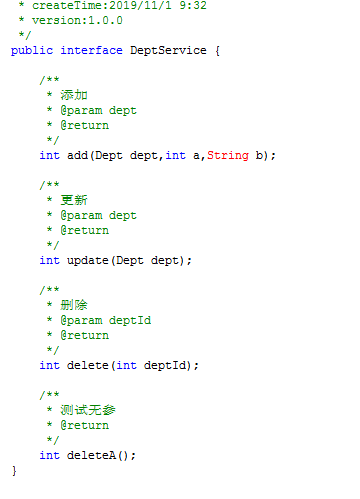
**环绕通知和前置通知，后置通知有着很大的区别，主要有两个重要的区别：**

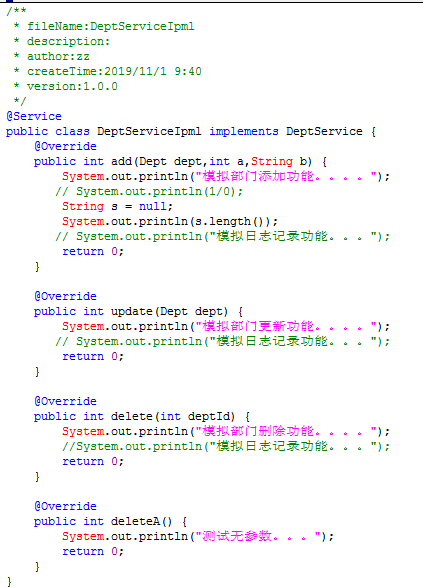
1） 目标方法的调用由环绕通知决定，即你可以决定是否调用目标方法，而前置和后置通知 是不能决定的，他们只是在方法的调用前后执行通知而已，即目标方法肯定是要执行的。

2） 环绕通知可以控制返回对象，即你可以返回一个与目标对象完全不同的返回值，虽然这很危险，但是你却可以办到。而后置方法是无法办到的，因为他是在目标方法返回值后调用

**3.5 AOP注解**

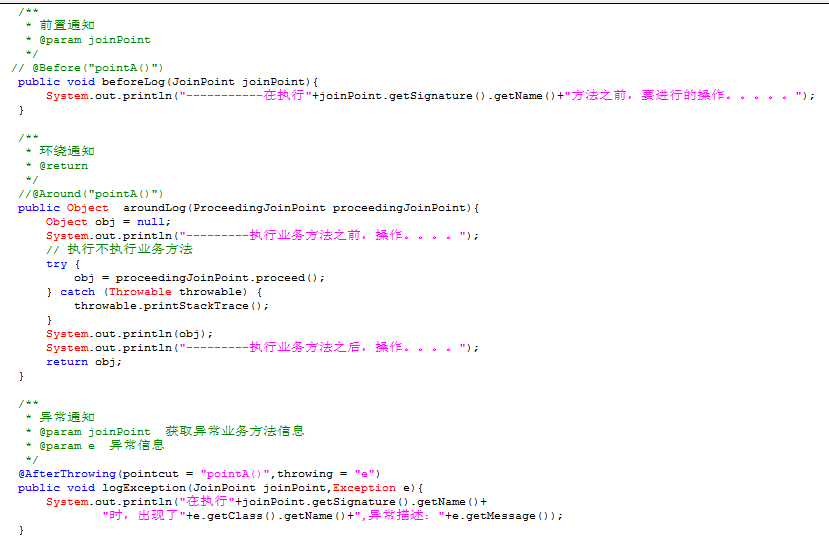
1，接口和实现类

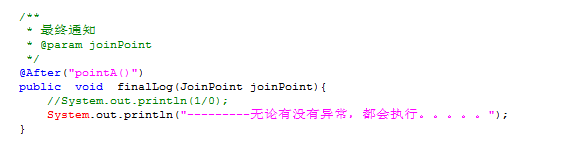




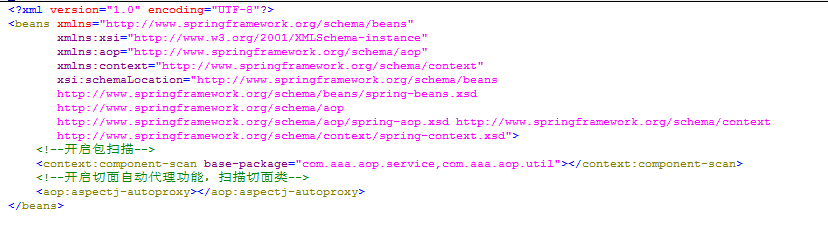
2，日志工具类



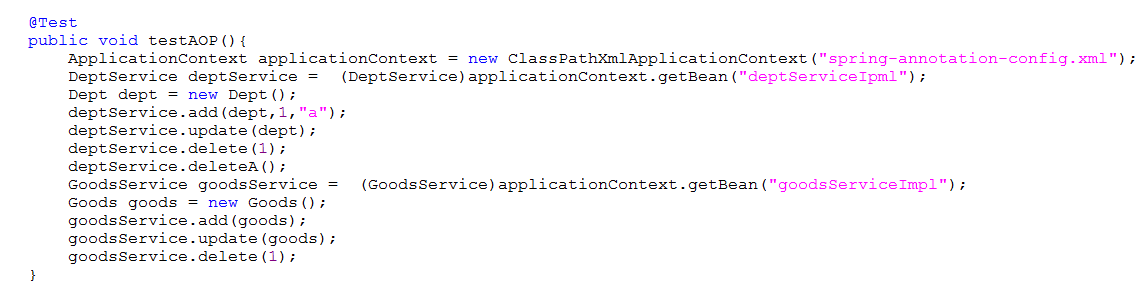




3，配置



4，测试



**4、本章总结**

**4.1 总结本章知识点**

**4.2 面试题答案**

**4.3 预习下一章重点**