**Spring加强**

## Spring中的概念

### 一．Spring控制反转（IOC）

Spring控制反转就是将对象中对象的实例化过程交给Spring框架来完成，Spring的控制反转实际上就是依赖注入（DI dependency injection）实现的，依赖注入是控制反转的一种实现形式。

### 二．Spring在进行对象初始化时，对象的初始化顺序

Class A{

@Autowired

B b;

}

Class B{

@Autowired

C c;

}

Class C{

}

结论：在初始化一个对象时，该对象依赖链的所有对象，都先会在IOC容器中初始化，然后再赋值给该对象。

IOC容器：用于存储初始化后的Bean对象。

### 三．面向切面（AOP）

AOP是一个思想，它的最终目的就是实现程序的解耦。与其说是面向切面编程，还不如说是面向规则编程，只有统一了各模块之间的编写的规则，最终才能将程序组装到一起运行。

AOP的核心就是切面，它将那些影响多个类的行为封装到可重用的模块当中。

Spring中面向切面思想的运用一个最大的体现就是Spring的事务管理，在Service层我们只需要将Service配置好Spring事务，Spring会自动在执行每个方法时开启事务，方法结束后在关闭事务。

AOP和IOC都运用了模块化方式将程序中复杂的开发问题变得更为简单，**实现了程序的解耦**。

## 俯瞰Spring架构设计

1. Spring中常用的设计模式
2. 代理模式
3. 工厂模式
4. 单例模式
5. 委派模式
6. 策略模式
7. 原型模式