目录

[Spring概述 1](#_Toc108736670)

[Spring3优点 1](#_Toc108736671)

[Spring开发代码的对比 2](#_Toc108736672)

[Springioc 4](#_Toc108736673)

[spring中的事务 4](#_Toc108736674)

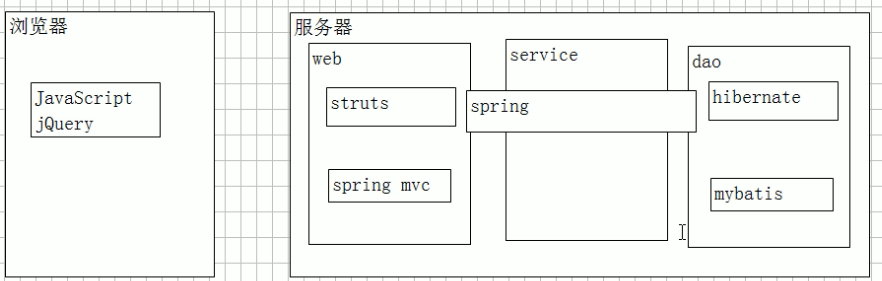
[Spring中的DI 5](#_Toc108736675)

[Spring boot 5](#_Toc108736676)

# Spring概述

**核心容器：beans、core、context、expression**

Mvc



Springmvc是spring里面的一个模块

## Spring3优点

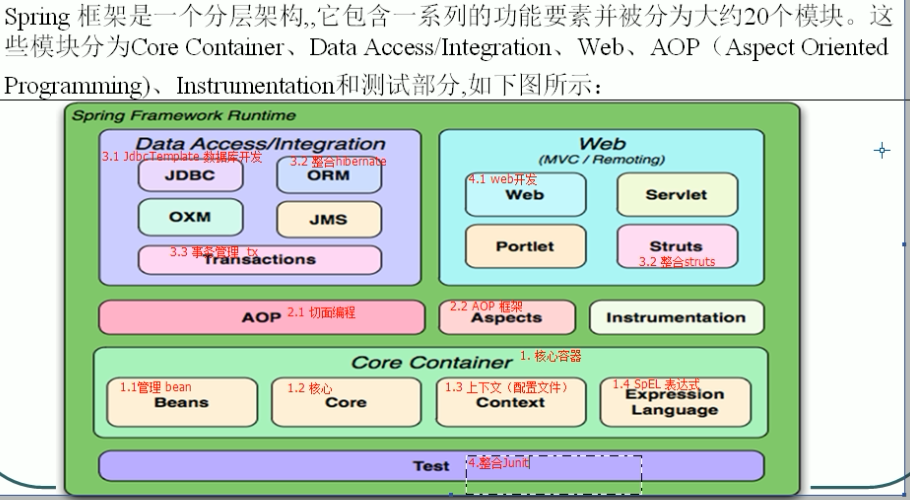
控制反转（IoC）对象创建和对象之间关系维护的控制权由程序员变为spring管理

（DI(依赖注入)用来描述spring中对象和对象之间依赖关系的创建）

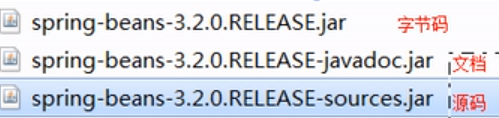
和面向切面（AOP）

* 方便解耦，简化开发 （高内聚低耦合）
  + Spring就是一个大工厂（容器），可以将所有对象创建和依赖关系维护，交给Spring管理
  + spring工厂是用于生成bean
* AOP编程的支持
  + Spring提供面向切面编程，可以方便的实现对程序进行权限拦截、运行监控等功能
* 声明式事务的支持
  + 只需要通过配置就可以完成对事务的管理，而无需手动编程
* 方便程序的测试
  + Spring对Junit4支持，可以通过注解方便的测试Spring程序
* 方便集成各种优秀框架
  + Spring不排斥各种优秀的开源框架，其内部提供了对各种优秀框架（如：Struts、Hibernate、MyBatis、Quartz等）的直接支持
* 降低JavaEE API的使用难度
  + Spring 对JavaEE开发中非常难用的一些API（JDBC、JavaMail、远程调用等），都提供了封装，使这些API应用难度大大降低

框架

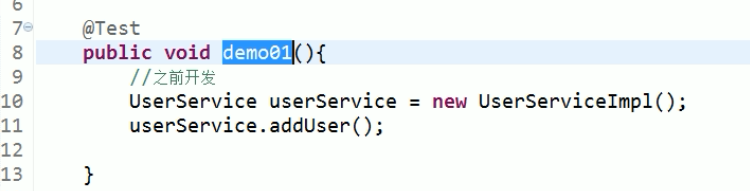


**导入jar包**



## Spring开发代码的对比

之前开发

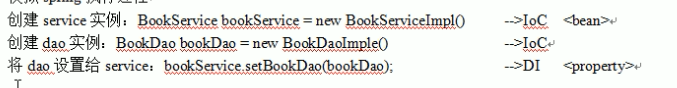


现在开发使用了管理bean

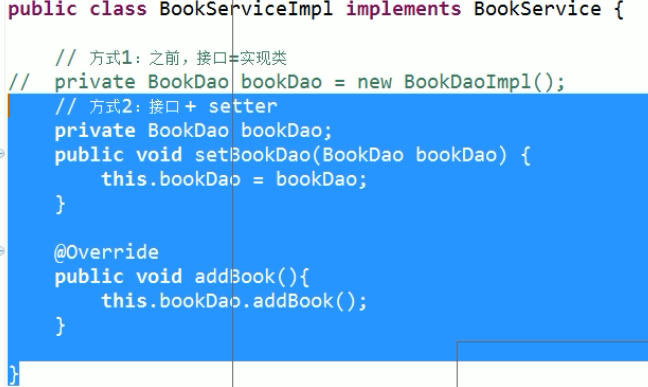
从配置文件中读取并使用



控制反转可能是由父类控制子类



方式一是老方法 方式二新方法



读取配置文件关键方法

延迟加载是只有第一次加载getbean的时候才会被实例化



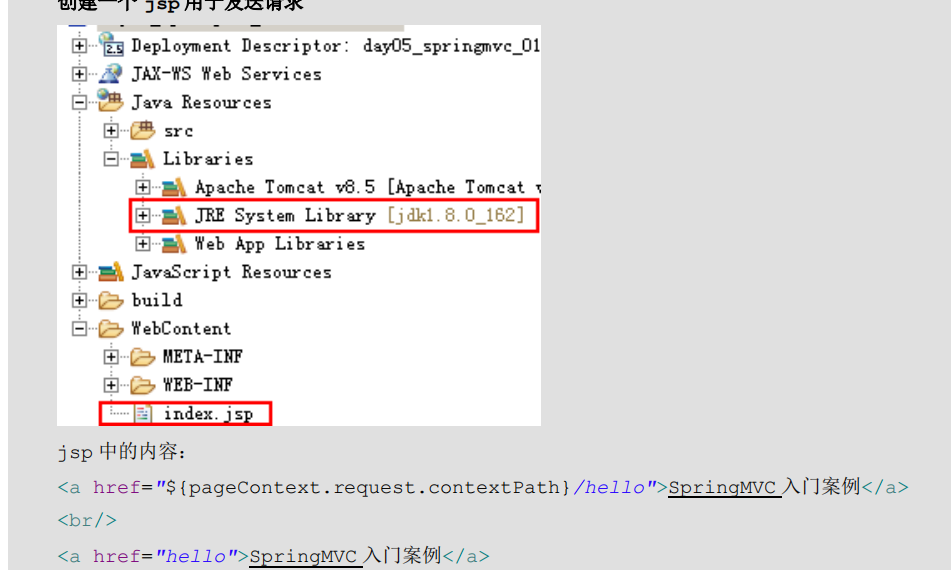
配置文件被加载，紧接着对象实例化，接口一定要找实现类

cglib字节码增强

 Java字节码增强指的是在Java字节码生成之后，对其进行修改，增强其功能，这种方式相当于对应用程序的二进制文件进行修改。Java字节码增强主要是为了减少冗余代码，提高性能等

Spring核心功能大概是生成bean然后交给spring代理来实现低耦合

Jsp或者html展示页面一般放在WEB-INF文件夹下面



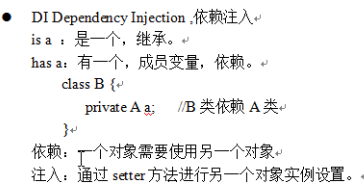
# Springioc

spring的ioc指的是把创建对象的权力交给IOC来

# spring中的事务

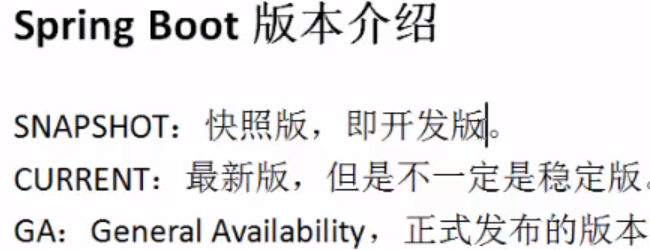
Spring中的事务是要实现复杂的业务需求逻辑的事务

# Spring中的DI



# Spring boot

不用像spring一样频繁的添加依赖，不用配置xml，如果还有需要也是通过配置类来

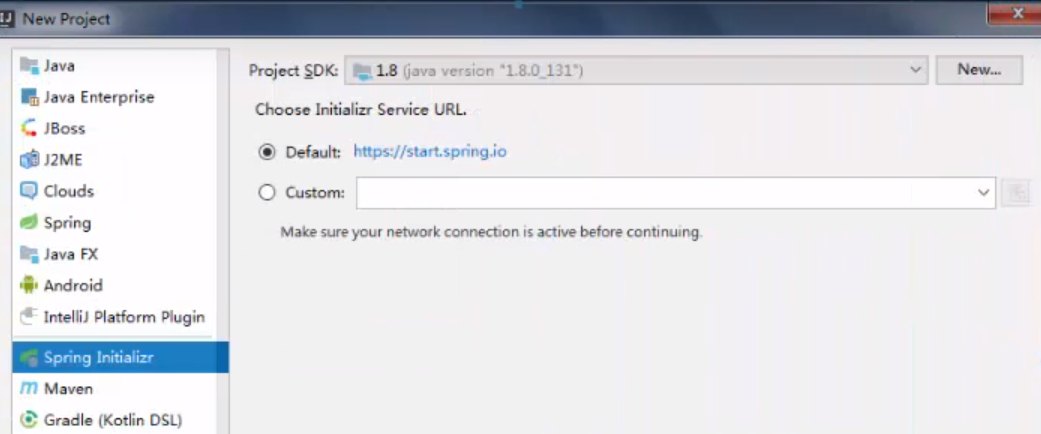


创建的三种方式

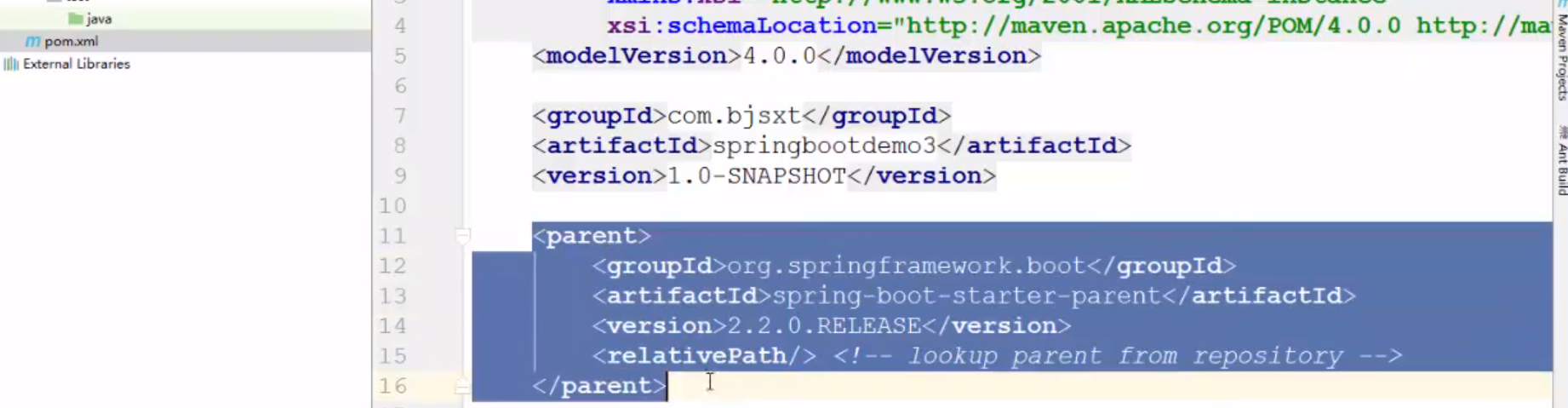
1.官方网站创建下载，idea导入



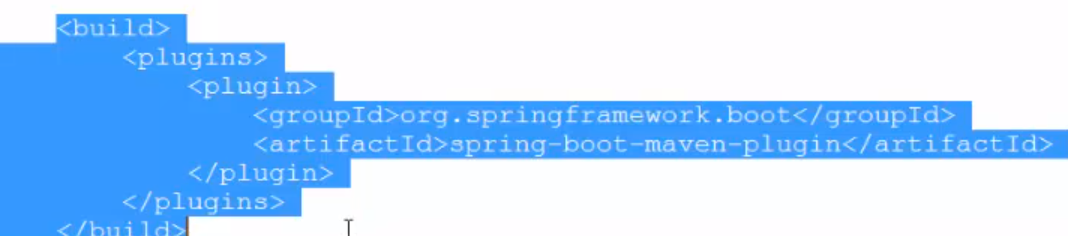
* 1. 脚手架工具，（本质上也是从官网下载）



3．通过maven，必须要在xml文件中导入以下配置



打包插件



Pom文件 继承 依赖 插件

Spring启动类

