Spring

Spring Framework:

是一个开源的应用程序框架， 提供了多种功能：依赖注入DI、面向切面编程aop、声明式事务管理TX，其他spring框架都是基于Spring Framework.

功能模块：

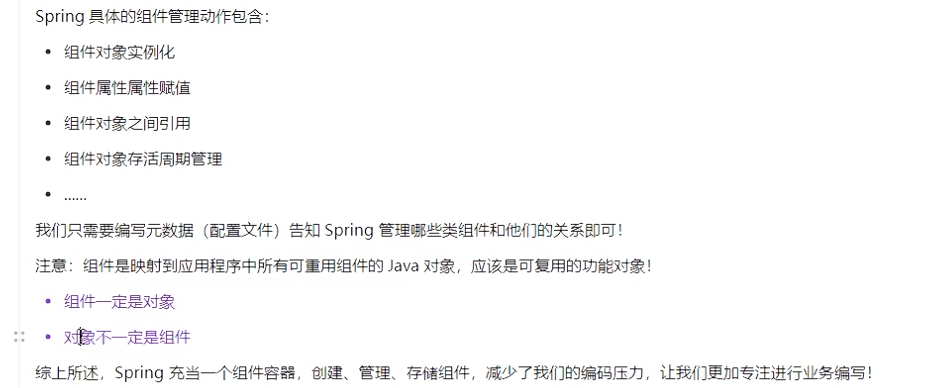
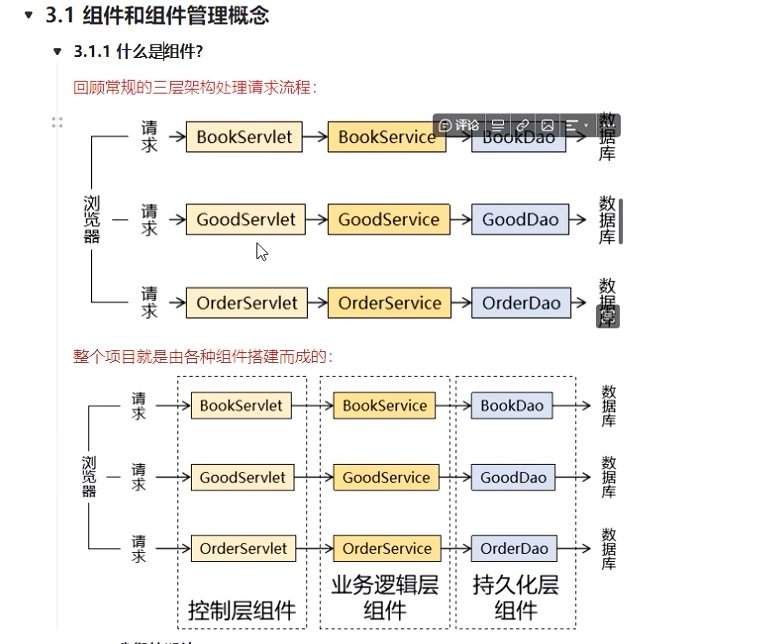
1核心容器（Core Container）：.

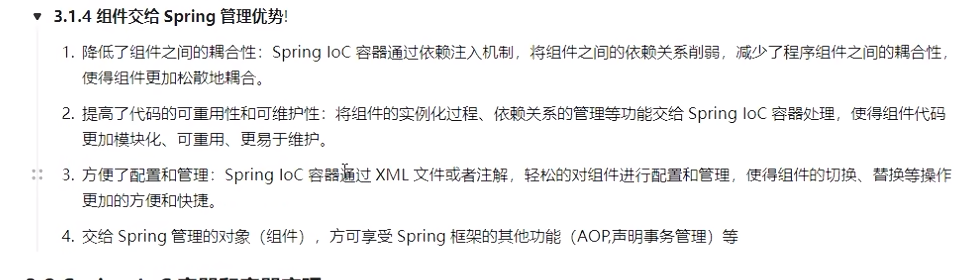
在Spring任何框架都必须使用核心容器

1. 面向切面编程：
2. TX:声明式事务管理
3. Spring MVC:提供了面向web应用程序的集成功能

组件：

组件就是多次使用且交给核心容器管理的对象





容器：

分为普通容器和复杂容器，普通程序只能对数据、对象进行简单的存储数组和集合就是普通容器，而复杂容器能对对象进行创建，赋值等一系列操作。

Spring IOC就是复杂容器。



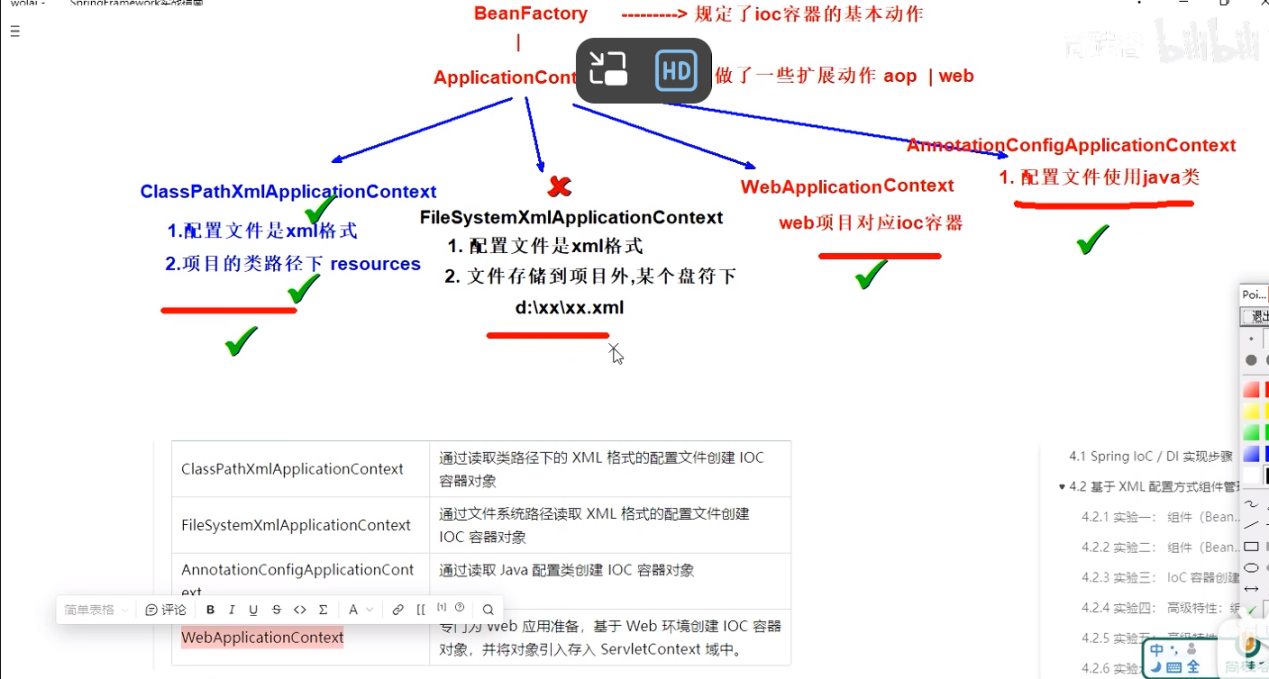
如何创建Spring IOC容器：

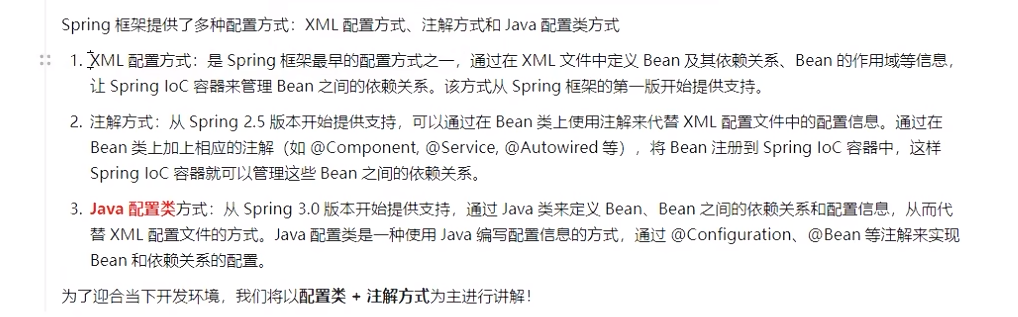
Spring IOC容器的实现接口：

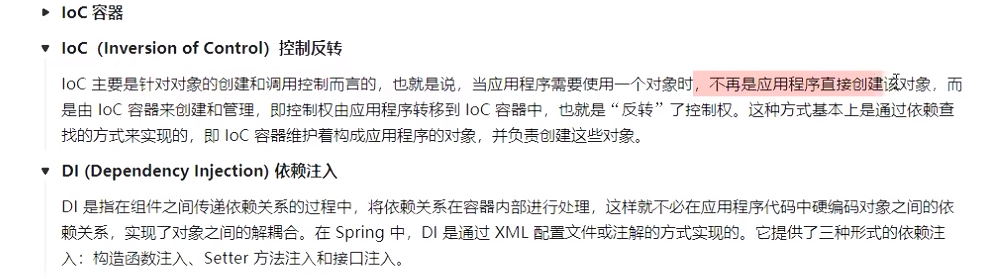
顶级父类接口BeanFactory,它定义了一些简单的容器功能，ApplicationContext接口实现了BeanFactory，在其基础上增加了一些功能。

Spring IOC容器的实现类：

1. ClassPathXmlApplicationContext:读取类路径下的XML配置文件信息创建IOC容器
2. FileSystemXmlApplicationContext:读取系统路径下XML配置文件信息创建IOC容器
3. AnnontationConfigApplicationContext:读取Java配置类创建IOC容器
4. WebApplicationContext:专门为Web程序准备基于Web环境创建IOC容器





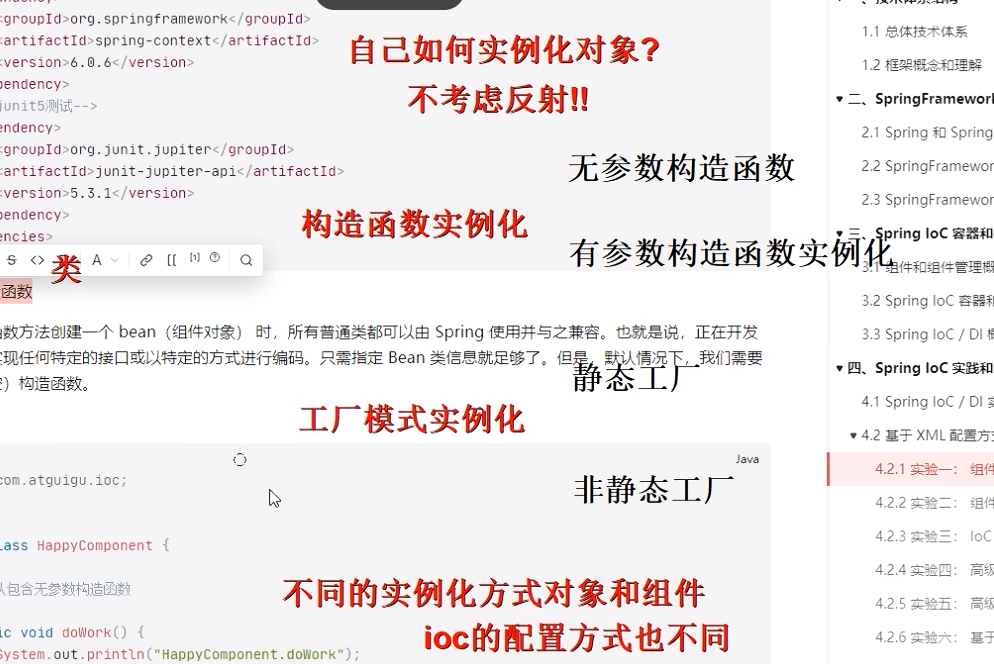


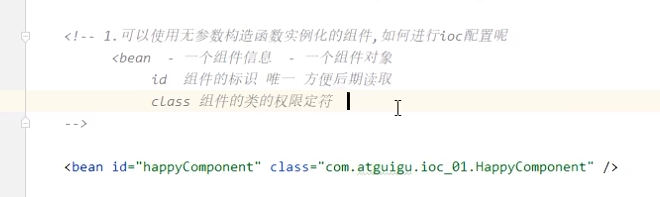
两种对象实例化方式：

构造函数实例化

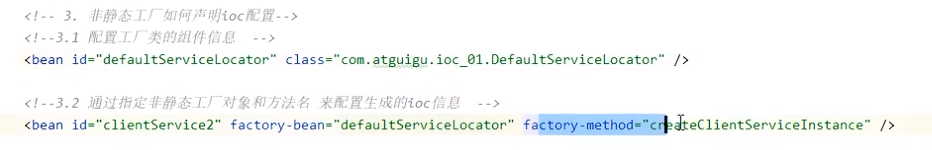
工厂模式实例化（通过类中的静态方法或非静态方法获取对象）

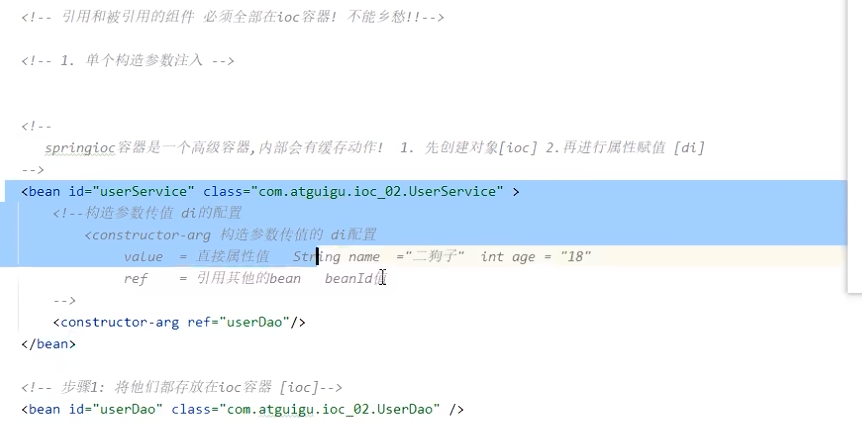
通过不同方式创建的对象和组件IOC配置文件信息也不一样



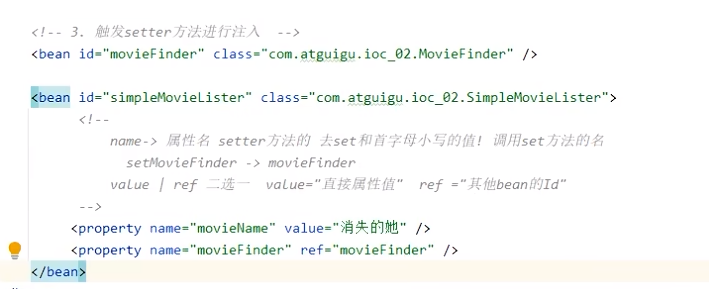












IOC容器功能总结：控制反转IOC（将本属于），依赖注入DI

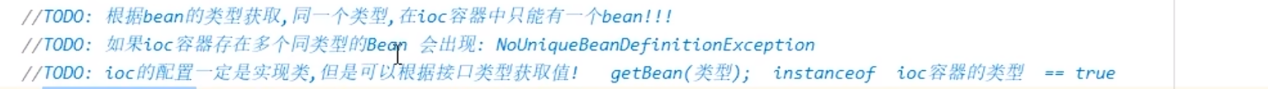
IOC容器的创建：

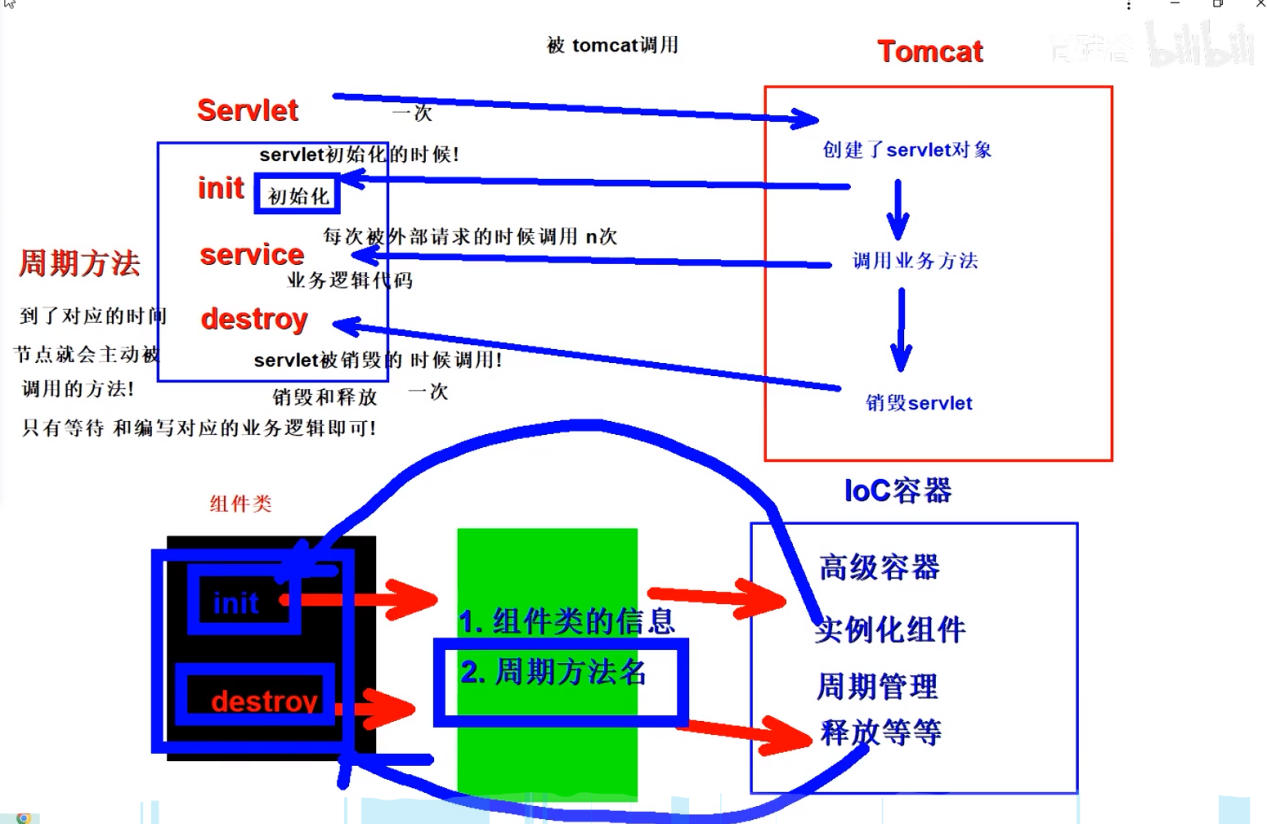
直接new对象形参传入IOC配置文件的名称











Bean的作用域：

每一个Bean在IOC容器中创建的个数由Bean的作用域Scope属性值决定，默认是singleton单例每个类在IOC容器中只会创建一个对象。当设置Scope的属性值为prototype则为多例，每使用一次该类就会单独创建一个对象在IOC容器中。



FactroyBean:

用于配置复杂的bean对象，实现FactroyBean接口，并重写接口中的方法，将复杂的创建对象的过程封装在getObject( )中，在getObject( )方法中返回对应的bean对象



基于注解配置IOC组件：

相关注解：

1. @Component（仅仅标注该类为IOC组件）
2. @Controller（标识该类为Controller请求处理类，并将该类放入ioc容器）
3. @Service（标识该类为逻辑处理类，并将该类放入IOC容器）
4. @Repository（标识该类为数据访问层，并将该类放入IOC容器）

2，3，4注解功能和1注解一样，不过起了标识作用

要让对应的注解生效还要配置包扫描，在XML配置文件中配置。

1.常规配置：

<context:component-scan base-package=”要扫描的包”/>会扫描对应包及其自包

1. 指定排除某个类

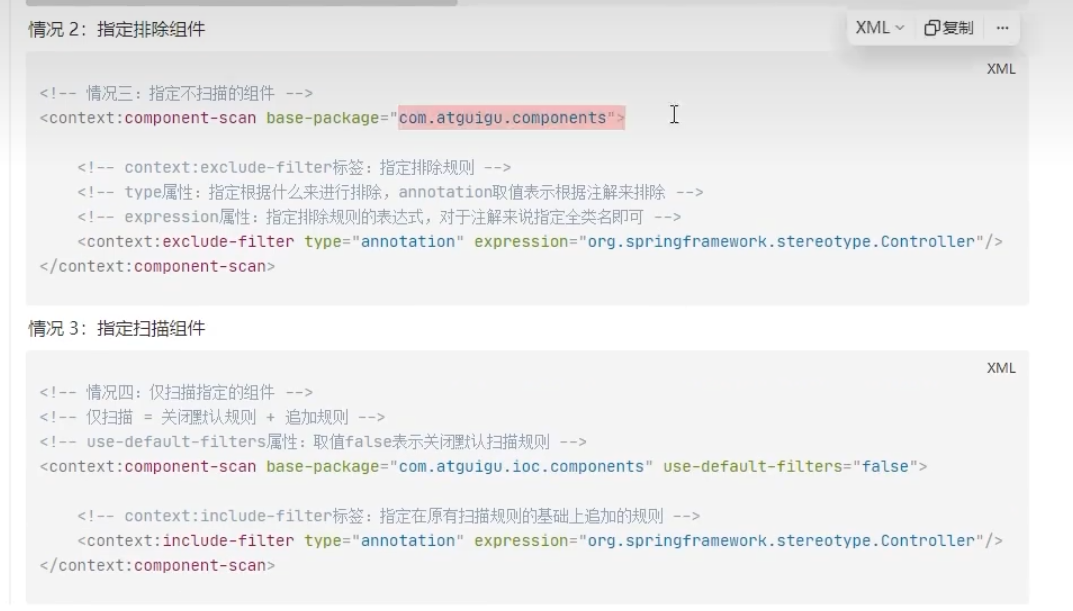
<context:component-scan base-package=”扫描的包”>

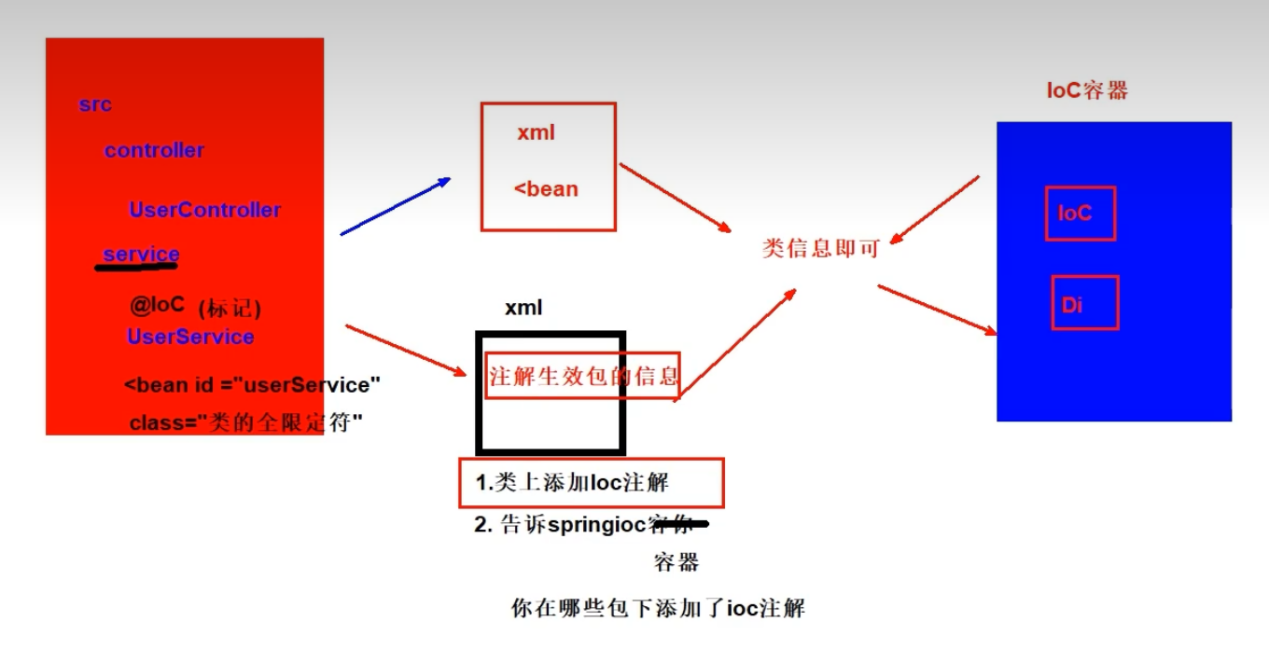
<context:exclude-filter type=”annotation” expression=”要排除包的全类名”/>

</context:component>

加上注解后相当于<bean id=”默认类的首字母小写” class=”就当前注解标注的类”

也可以在注解后加上括号里面指定value的属性值就可以指定bean的id







基于注解的周期方法声明：

在对应方法上加上@PostConstruct用来指定初始化方法，@PreDestroy指定销毁方法



基于 注解的DI依赖注入：

@Autowired







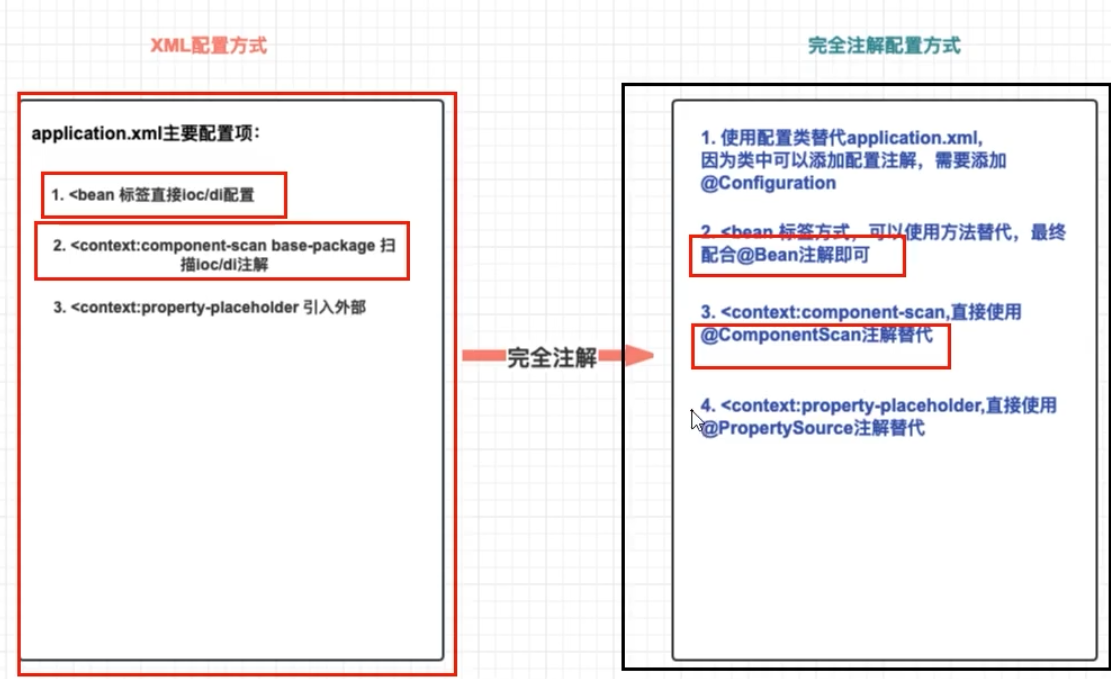
@value注解的作用：

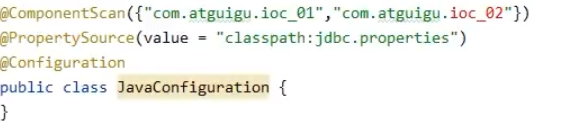
获取作用域中的值EL表达式



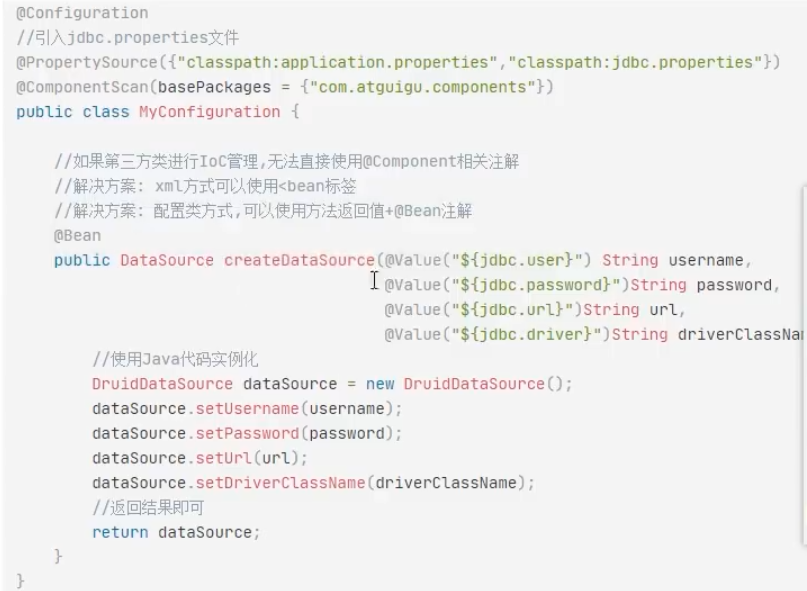
配置类和完全注解开发：

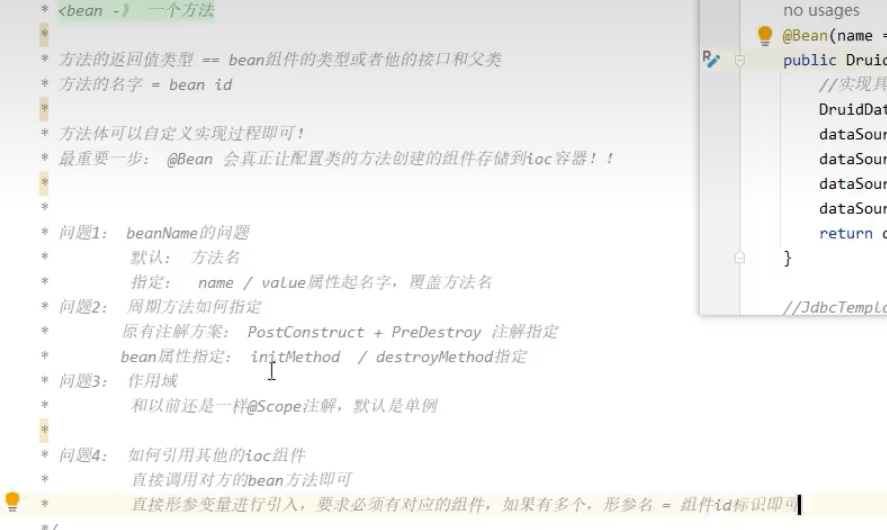
在对应类上加上@Configuration声明该类为配置类，再加上@ComponentScan注解声明要扫描的包，引入配置文件用@PropertySource，要配置第三方类在对应的方法上加上@bean注解将该类放入IOC容器





bean注解和value注解的具体使用





@import

作用:当有多个配置文件时可以将多个配置文件导入同一个文件统一管理

Spring-Test:

测试环境的搭建。简化每次测试都需要创建对应的IOC容器，在测试类上加上注解@SpringJUnitConfig(value = 配置类.class)或(location = 指定的配置文件xml)

value用于指定配置类，location用于指定对应的xml配置文件。

注解配置完成后需要对应的对象直接注入即可。

不过需要导入对应的依赖

