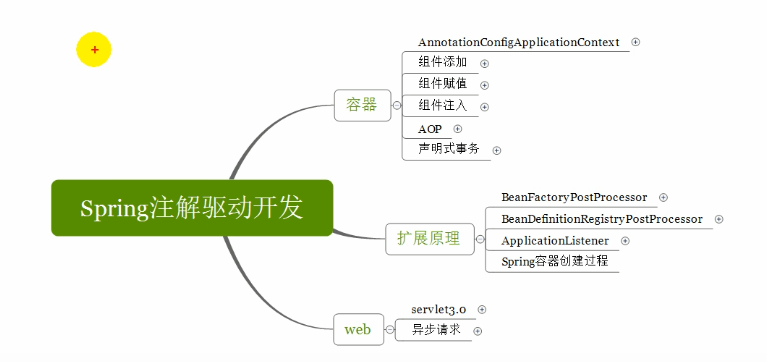
**Spring源码解析**

**2019.7.2**

参考网址：[Spring源码解析](https://www.bilibili.com/video/av32102436?from=search&seid=4114505841510092049)



[mvnrepository仓库](https://mvnrepository.com)

**一、组件注册**

1、@ComponentScan—自动扫描组件&指定扫描范围

2、@Scope—设置组件作用域

对于bean来说的，组件的作用域@Scope：调整作用域

参数有四个：

prototype：多实例的：IOC容器启动并不会调用方法创建对象放在容器中。每次获取的时候才会调用方法创建对象

singleton：单实例的（默认值）：IOC容器启动会调用 方法创建对象放到IOC容器中。以后每次获取就是直接从容器（map.get()）中拿.

request：同一次请求创建一个实例

session：同一个session创建一个实例

3、@Lazy—bean懒加载

单实例bean，默认在容器启动的时候创建对象；

懒加载：容器启动不创建对象。第一次使用（获取）bean是才创建对象，并初始化

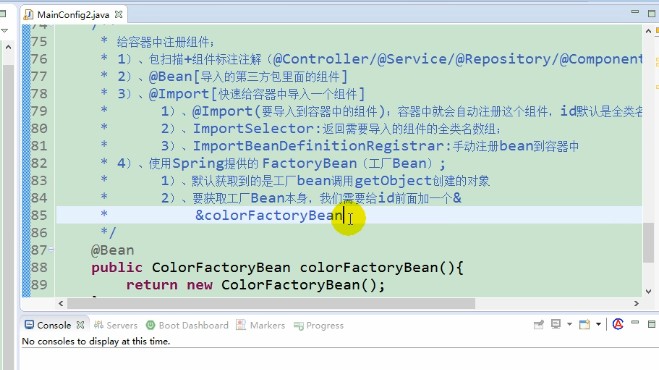
4、@Conditional—按照条件注册bean

按照一定的条件进行判断，满足条件给容器中注册bean

一种是在方法（配置类）中，一种是在类（配置类）中组件统一设置（满足当前条件，这个类中配置的所有bean注册才能生效）。

**小结：**

给容器中注册组件：

（1）包扫描+组件标注注解（@Controller/@Service/@Repository/@Component）[自己写的类]

（2）@Bean[导入的第三方包里面的组件]

（3）@Import[快速给容器中导入一个组件]

1）@Import（要导入到容器中的组件），容器中就会自动注册这个组件，id默认是全类名

2）ImportSelector：返回需要导入的组件的全类名的数组【springboot用的多】

3）ImportBeanDefinitionRegistrar：手动注册bean到 容器中

（4）使用Spring提供的FactoryBean（工厂bean）

1）默认获取到的是工厂bean调用getObject创建的对象

2）要获取工厂bean本身，我们需要给id前面加一个&

&colorFactoryBean

**二、Bean的生命周期**

Bean的生命周期：bean创建—初始化—销毁的过程

容器管理bean的生命周期：

我们可以自定义初始化和销毁方法：容器在bean进行到当前生命周期的时候来调用我们自定义的初始化和销毁方法

构造（对象创建）：

单实例：在容器启动的时候创建对象

多实例：在每次获取的时候创建对象

BeanPostProcessor.postProcessBeforeInitialization

初始化：

对象创建完成，并赋值好，调用初始化方法

BeanPostProcessor.postProcessAfterInitialization

销毁：

单实例：容器关闭的时候

多实例：容器不会管理这个bean；容器不会调用销毁方法

1、@Bean指定初始化和销毁方法

通过@Bean指定init-method和destroy-method

2、通过让Bean实现InitializingBean（定义初始化逻辑），DisposableBean（定义销毁逻辑）

3、我们也可以使用JSR250

@PostConstruct：在Bean创建完成并且属性赋值完成，来执行初始化方法

@PreDestroy：在容器销毁bean之前通知我们进行清理工作

4、BeanPostProcessor【interface】：bean的后置处理器

在bean初始化前后进行一些处理工作

postProcessBeforeInitialization：在初始化之前工作

postProcessAfterInitialization：在初始化之后工作

Spring底层对BeanPostProcessor的使用：

Bean赋值，注入其他组件，@Autowired，生命周期注解功能，@Async,xxx BeanPostProcessor