1. 体系

Org.pringframework.context.support.AbstractApplicationContext#refresh

Org.pringframework.context.support.AbstractApplicationContext#finishBeanFactoryInitialization

Org.springframework.beans.factory.config.ConfigurableListableBeanFactory#preInstantiateSingletons

Org.springframework.beans.factory.support.AbstractBeanFactory#getBean(string)?实例化bean调用getBean

Org.springframework.beans.factory.support. AbstractBeanFactory#doGetBean

如果是创建对象，这里大部分返回null---循环引用，第一次调用getSingleton

Org.springframework.beans.factory.support.DefaultSingletonBeanRegistry#getSingleton( java)

第二次调用getSingleton 单例池拿不到 ，先添加到beforeSingletonCreation集合当中

Org.springframework.beans.factory.support.AbstractBeanFactory#createBean

以上步骤是来实例化bean ---spring bean object

Org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory

创建了一个beanWrapper需要创建的对象，没有执行自动装配----lifecycle ca

Org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory

备注：spring 对象实例化和生命周期在AbstractAutowireCapableBeanFactory 中完成----方法createBeanInstance 500行左右的源代码 ：final Objcet bean = instanceWrapper.getWrappedInstance();

BeanFactoryPostProcessor

后置处理器：

InstantiationAwareBeanPostProcessor ---- postProcessBeforInstantiation

SmartInstantiationAwareBeanPostProcessor ------determineCandidateConstructors

MergedBeanDefinitionPostProcessor -------postProcessMergedBeanDefinition

SmartInstantiationAwareBeanPostProcessor ------getEarlyBeanReference

InstantiationAwareBeanPostProcessor -------postProcessAfterInstantiation

InstantiationAwareBeanPostProcessor -------postProcessPropertyValues

BeanPostProcessor ------- postProcessBeforeInitialization

BeanPostProcessor ------- postProcessAfterInitialization

[www.cnblogs.com/CodeBear/p/10374261.html](http://www.cnblogs.com/CodeBear/p/10374261.html)

1. getSingleBean

循环引用问题

单例池拿不到，判断------到缓存池再拿 ，

2.

生命周期的回调其实是：Initialization 类下的方法：afterPropertiesSet() 方法

3.spring如何new 出一个对象？ 当一个类中有多个构造时

Eg：会根据构造参数最多那个new出对象来

通过反射的filed

4. 自动注入模型：spring注入一个类的技术有两种方式：即ByName ，ByType

4.1 注入方式有4种：service.setAutowireMode( 1 );

A．装配模型：0 （默认是0 ）

B．

Filed.set( orderService)

装配模型：1

service.setAutowireMode( 1 ); 为1 时如下：

New OrderService ---自动注入----byname-🡪 writeMethod（规则—以set开头 ---🡪 set方法里面的参数是在spring容器当中

. set方法的名称必须跟

）

C．装配模型：3 service.setAutowireMode( 3 ); 为3 时注入方式是： 如下：

New OrderService ---自动注入----3 ---根据构造方法自动装配，但set方法里面的参数是在spring容器当中

会把类里面所有带Set的方法找出来

1. 学习步骤
2. 管网：<https://spring.io/projects>

2.

**三．5的新特性 重点**

**1.** [**Spring WebFlux**](https://docs.spring.io/spring/docs/current/spring-framework-reference/web-reactive.html)