**春天的杰作Spring源码分析**

主讲：Cat 老师

北京动力节点教育科技有限公司  
2020 • 北京

动力节点•版权所有•禁止传播

## 快速回顾

1. 源码构建

环境、版本IDEA2020.1.2，建议和课堂上的保持一致，其他版本下可能会遇到环境错误；

插件的安装；

Gradle乱码的处理；

在D:\dev\ideaIU\bin\idea64.exe.vmoptions文件最后加上

-Dfile.encoding=UTF-8

在C:\Users\Administrator\AppData\Roaming\JetBrains\IntelliJIdea2020.1\idea64.exe.vmoptions文件最后加上

-Dfile.encoding=UTF-8

1. IOC容器源码剖析

## IOC源码分析

1. ApplicationContext context = new AnnotationConfigApplicationContext(SpringConfig.class);

2、new AnnotationConfigApplicationContext(SpringConfig.class);--> {

1、this();-->{

//spring内部加载了5个BeanDefinition：就是对bean的描述;

1、this.reader = new AnnotatedBeanDefinitionReader(this);

//放置了@Compment注解

2、this.scanner = new ClassPathBeanDefinitionScanner(this);

}

2、register(componentClasses); {

//把componentClasses(SpringConfig)变成bean的描述：BeanDefinition;

//然后把BeanDefinition包装成BeanDefinitionHolder;

//然后取出BeanDefinitionHolder里面的BeanDefinition;

//注册到Map<String, BeanDefinition> beanDefinitionMap中;

}

3、refresh();//参考下面的具体步骤

}

3、refresh(); {

//该方法总共有12个步骤，重点看第5步和第11步，宏观上的步骤是：

//根据项目中的bean（@Service、@Compment...注解的bean）得到BeanDefinition

//把BeanDefinition放入（注册）beanDefinitionMap中

//把beanDefinitionMap里面的BeanDefinition拿出来，依次创建所对应的对象；

}

}

## refresh()方法12个步骤分析

*// 1、预准备刷新上下文*

prepareRefresh();

*// 2、告诉子类去刷新内部的bean factory，如果是xml方式开发，此步骤很重要，如果是注解方式开发，这一步没有做什么操作；*ConfigurableListableBeanFactory beanFactory = obtainFreshBeanFactory();

*// 3、预处理BeanFactory，注册systemEnvironment，environment，systemProperties几个bean*prepareBeanFactory(beanFactory);

*//4、由子类覆盖该方法做额外的处理，默认spring框架什么也没有做*postProcessBeanFactory(beanFactory);

*//5、调用各种BeanFactoryPostProcessor，注解开发的时候，解析和扫描注解得到BeanDefinition，然后放入到BeanDefinition的Map中，这一步非常重要（核心步骤）*

invokeBeanFactoryPostProcessors(beanFactory);

*//6、注册BeanPostProcessors，用于拦截Bean的创建，如果没有bean processors，此步骤什么也不做*registerBeanPostProcessors(beanFactory);  
  
*//7、实例化注册messageSource的bean，不同语言环境信息，国际化处理*initMessageSource();  
  
*//8、在上下文初始化注册applicationEventMulticaster的bean，应用事件广播*initApplicationEventMulticaster();

*//9、初始化其他的bean，默认情况下Spring框架什么也不做*onRefresh();  
  
*//10、在所有bean中查找listener bean并注册到消息广播中，没有的话就什么也不做；*registerListeners();  
  
*//11、实例化所有非延迟初始化的单例bean对象，这一步非常重要（核心步骤）*

finishBeanFactoryInitialization(beanFactory);  
  
*//12、完成刷新过程，通知生命周期处理器lifecycleProcessor刷新，同时发出ContextRefreshEvent通知相关事件*finishRefresh();

在以上的12个步骤中，只有第5步和第11步是非常重要的，是我们源码阅读重点需关注的，其他步骤可以略读，下面我们分别剖析一下这两个步骤，首先剖析第5步：

## IOC容器12步中的第5步剖析

*//调用BeanFactoryPostProcessor*

*invokeBeanFactoryPostProcessors(beanFactory);*

*1、调用BeanFactoryPostProcessors*

*PostProcessorRegistrationDelegate.invokeBeanFactoryPostProcessors(beanFactory, getBeanFactoryPostProcessors());*

*2、调用PostProcessors，里面处理bean的定义  
invokeBeanDefinitionRegistryPostProcessors*(currentRegistryProcessors, registry);

*//3、BeanDefinition的注册*

postProcessor.postProcessBeanDefinitionRegistry(registry);

*4、处理配置类的bean的定义*processConfigBeanDefinitions(registry);

*5、解析每一个带有@Configuration注解的类*ConfigurationClassParser parser = new ConfigurationClassParser(...);

*//对配置类中注解进行处理*parser.parse(candidates);

*6、加载bean的定义: 把BeanDefinition放入到一个Map中*this.reader.loadBeanDefinitions(configClasses);

*7、注册bean的定义*this.registry.registerBeanDefinition(beanName, beanDefToRegister);

*8、注册bean定义，即把beanDefinition放入Map中*this.beanDefinitionMap.put(beanName, beanDefinition);

以上是第5步的主要流程，大家可以基于该流程反复断点调试跟几轮代码，潜移默化中提升源码的自我阅读能力，此能力需要坚持一个过程和一段时间，非一蹴而就；

## 基于注解的Spring IOC容器宏观总结

1、创建AnnotationConfigApplicationContext;

2、创建AnnotationConfigApplicationContext的时候会先创建DefaultListableBeanFactory;

3、创建好DefaultListableBeanFactory之后，解析和扫描类上的注解并将所有Bean封装成BeanDefinition对象，该BeanDefinition对象封装了Spring所有bean的相关定义描述信息；

4、对Spring配置的所有bean做实例化，根据BeanDefinition对象的信息，底层采用反射创建具体的bean对象；

5、将实例化的bean对象放入到一个Map中，后续应用程序就能从该Map中获取到bean对象，该Map也就是我们的ioc容器；

## 作业与复习

到目前，我们把IOC的核心流程代码已经分析差不多了，还差一个bean对象的实例化过程，如果让你现在写一个Spring的IOC，应该怎么实现，可以思考一下；