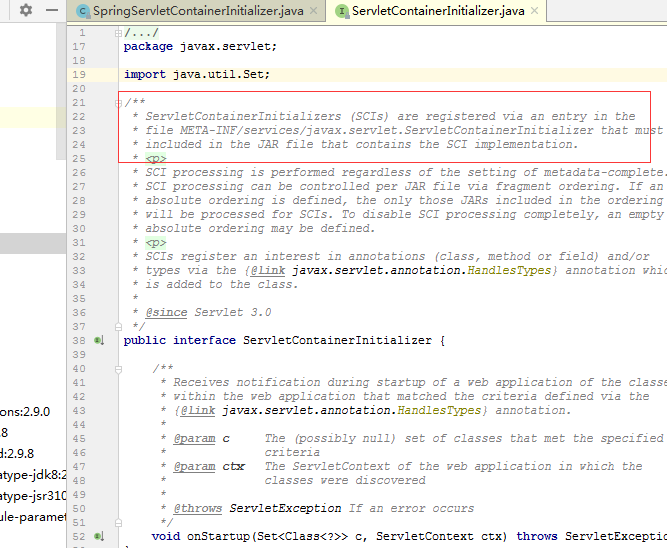
常规我们在开发javaweb应用时，都会使用web.xml文件，在该文件中添加我们的serverlet，filter，并创建spring容器的上下文

Serverlet升级到3.0后，支持无web.xml的开发，就是这个配置项全部使用java代码去配置。

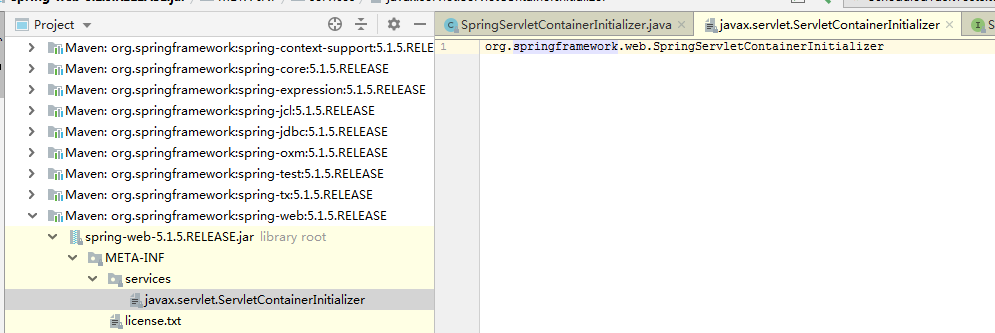
如何做的呢。想要实现无web.xml文件的开发，我们必须的实现ServletContainerInitializer接口，该接口由Tomcat提供的，该接口中只有一个onStartup

方法。我们自己去实现该方法。问题现在就来了，容器是如何知道我们有实现ServletContainerInitializer接口的实现类的呢，这个其实是我们告诉容器的，不过也是容器约定好的配置，我们将实现类的全类名放在配置中，容器去读取这个配置，就知道谁实现了ServletContainerInitializer接口。具体是配置呢。我们看代码：  


接口上的注释已经告诉我们在jar包的METE-INF/services目录下的javax.servlet.ServletContainerInitializer文件中，容器在启动时会去扫描每个jar包是否有包含META-INF/services/javax.servlet.ServletContainerInitializer文件，有的话就会读取配置，我们看看spring webmvc是如何实现该接口的，代码如下：



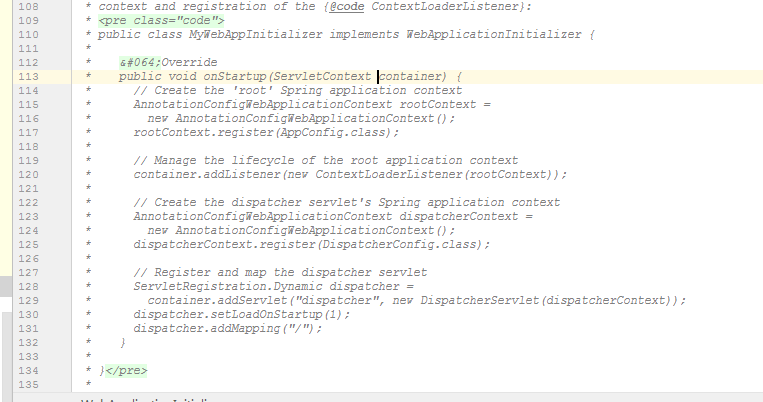
Jar包中的配置如下：



onStartup 方法的第一个参数表示我们可以用于处理的类，通过@HandlesTypes注解指定。也就是如下意思：

其中@HandlesTypes注解表示CustomServletContainerInitializer 可以处理的类，在onStartup 方法中，可以通过Set<Class<?>> c获取得到。

我们可以看到spring webmvc指定了WebApplicationInitializer.class 该类是spring提供的一个接口，我们自己实现。具体可以做哪些有我们自己确定，spring给了样例



在非springboot的应用下，我们自己继承AbstractAnnotationConfigDispatcherServletInitializer这里类就可以实现去除web.xml文件