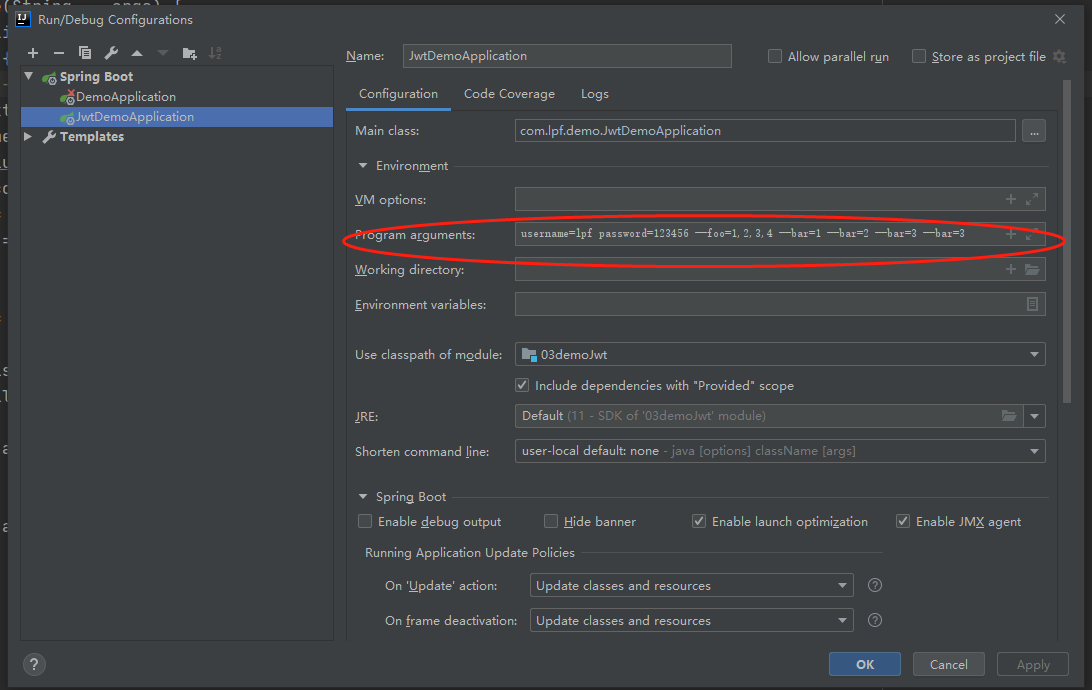
# Spring Boot 项目启动时参数传递

Springboot 项目启动时传递参数的方式

## Idea 启动

通过idea 启动的时候，在 Run/Debug Configurations 中配置要添加的参数



参数形式：

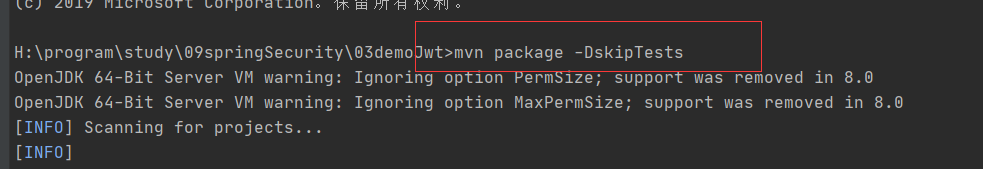
username=lpf password=123456 --foo=1,2,3,4 --bar=1 --bar=2 --bar=3 --bar=3

这里要添加的参数，分为两种形式，一种是以 – 开头的，另一种是直接以参数名开头的

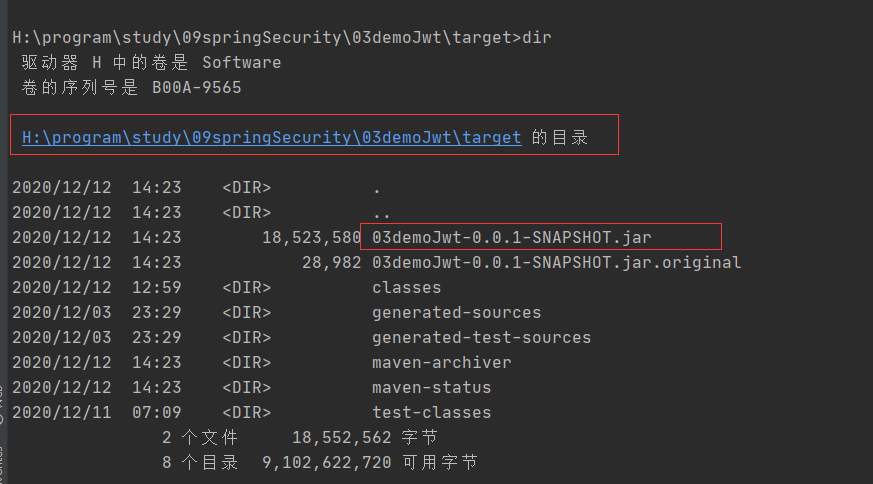
同时，可以向项目中传递集合：比如 bar 参数。

## java命令行启动

1. 先通过 mvn package –DskipTests 打包springboot 项目



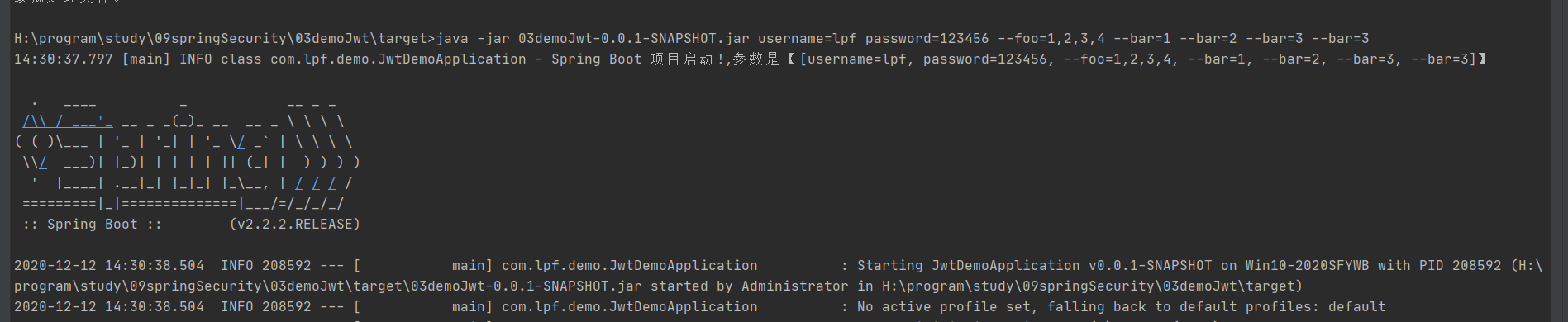
打包后的jar 包文件位于项目的target 目录下如图：



1. 执行

Java –jar \*\*.jar username=lpf password=123456 --foo=1,2,3,4 --bar=1 --bar=2 --bar=3 --bar=3命令来启动springboot 项目

如下，可以看到项目正常启动。

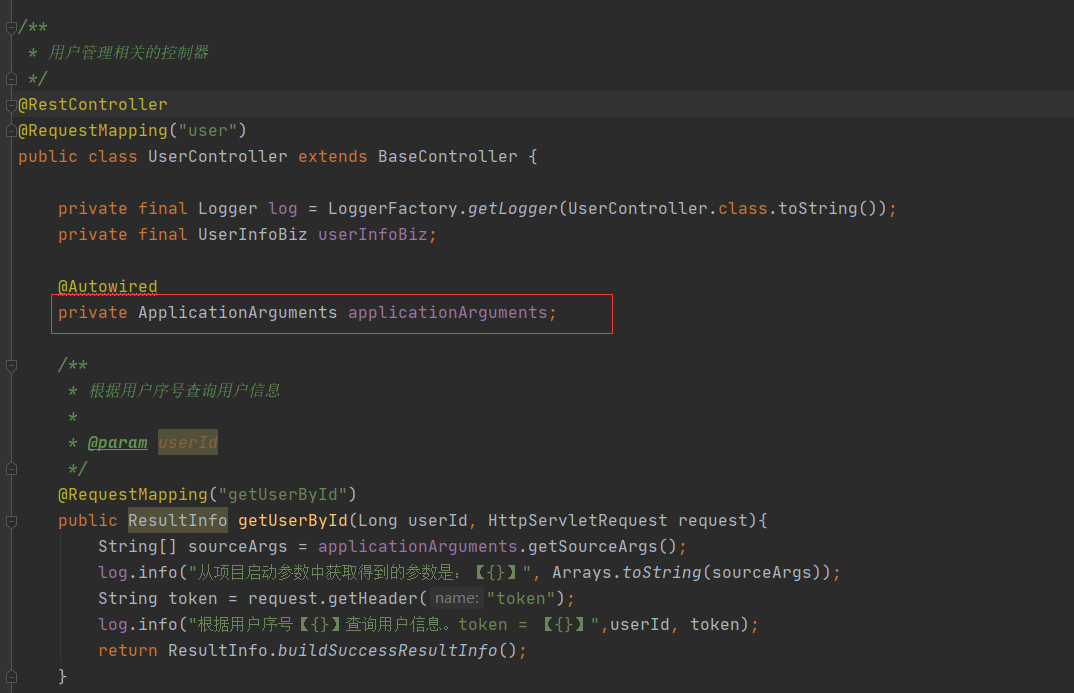


## 项目中使用参数

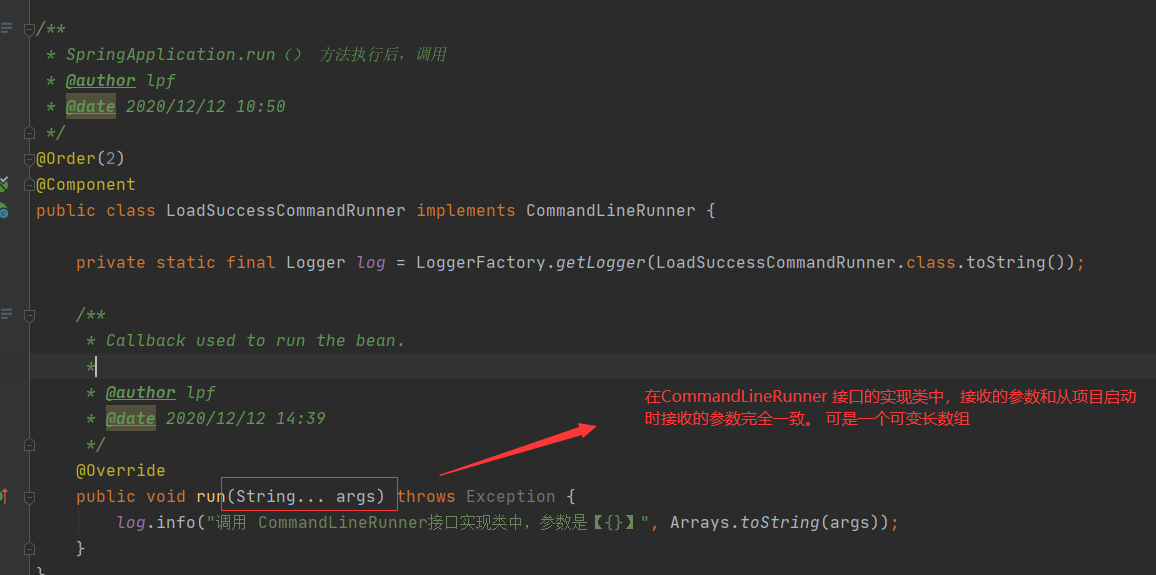
springBoot 项目启动过程中，会将参数封装到

### 在普通的bean中使用参数

通过将ApplicationArguments 这个类注入到bean中，之后可以从这个对象中获取到传入的参数。

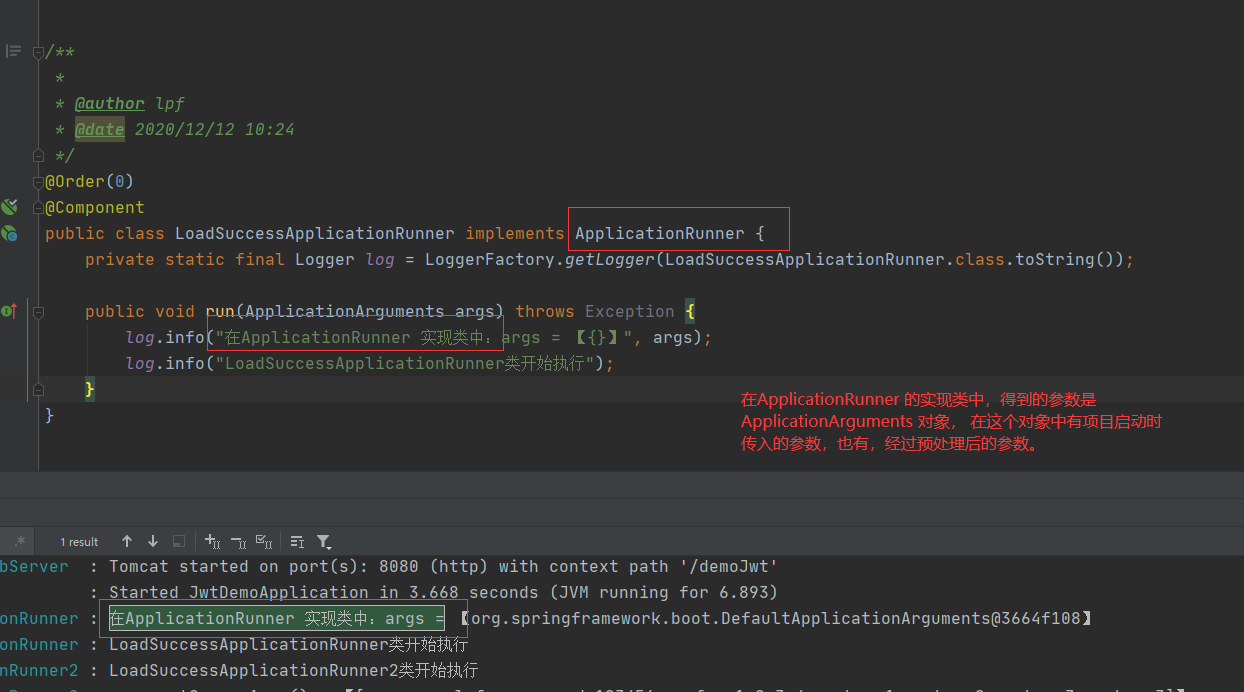


### 在CommandLineRunner 接口实现类中使用参数



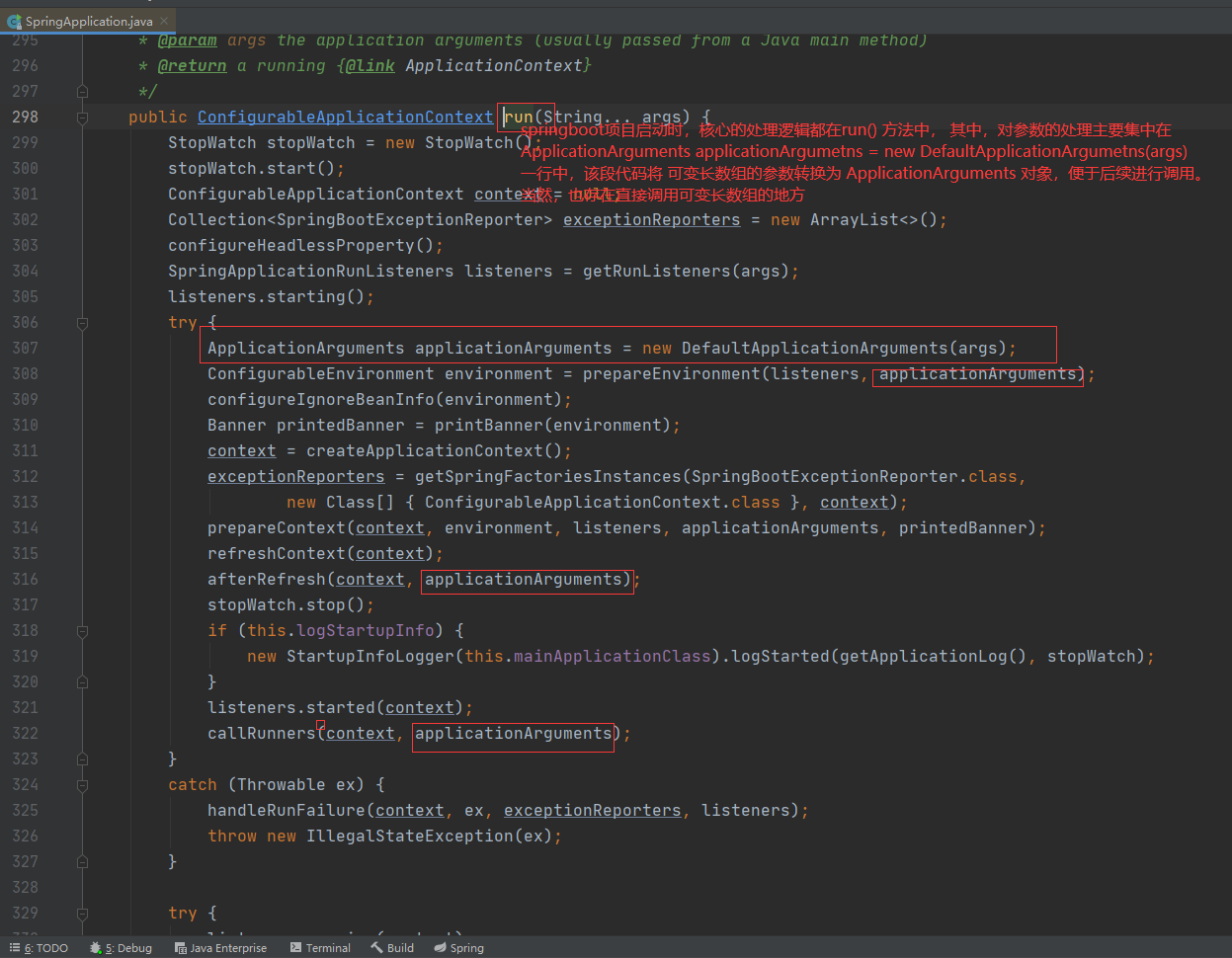


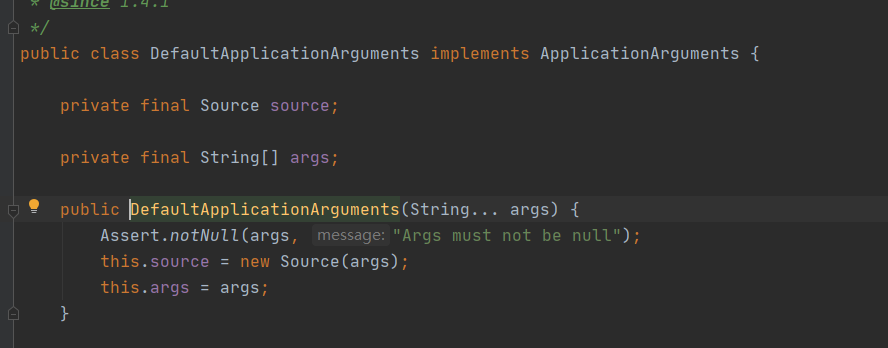
### 在ApplicationRunner 接口实现类中使用参



## Sprinboot源码解读

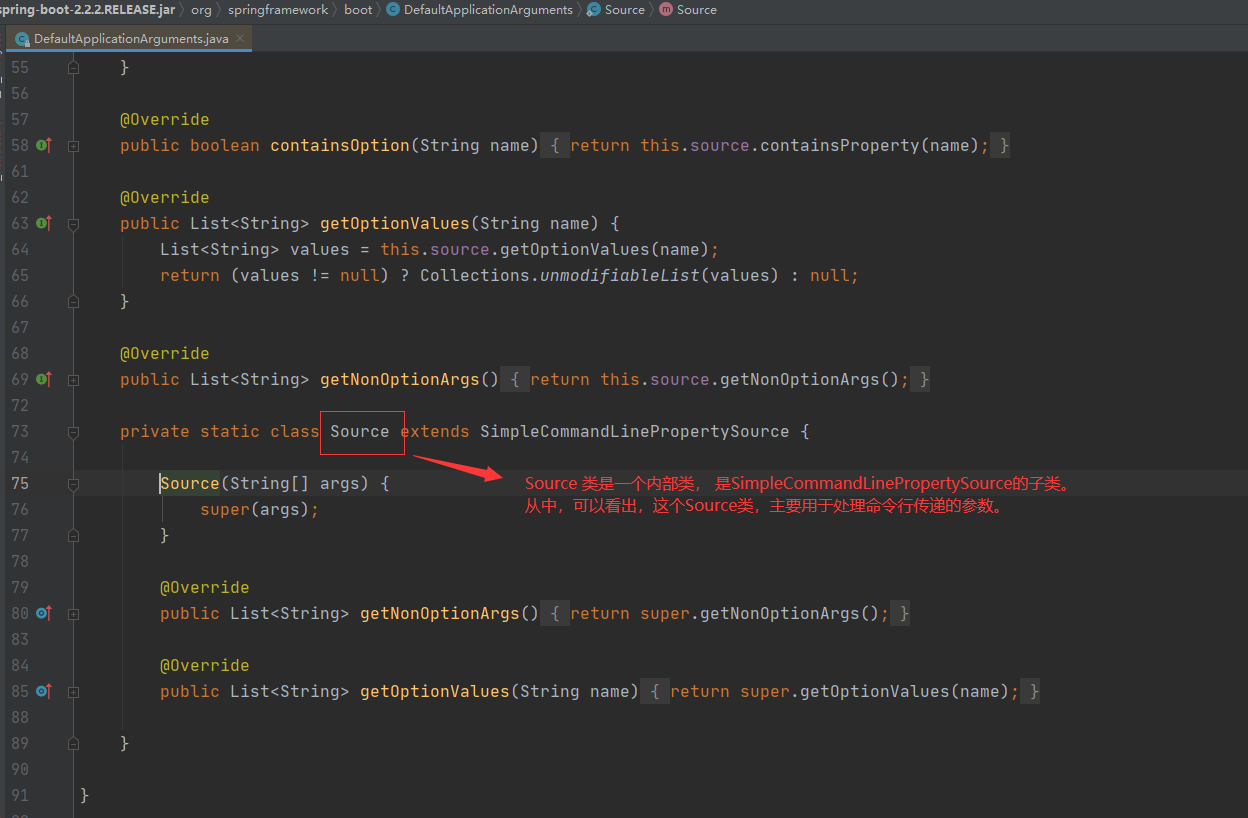
从源码中探索springboot 项目启动时对参数的处理



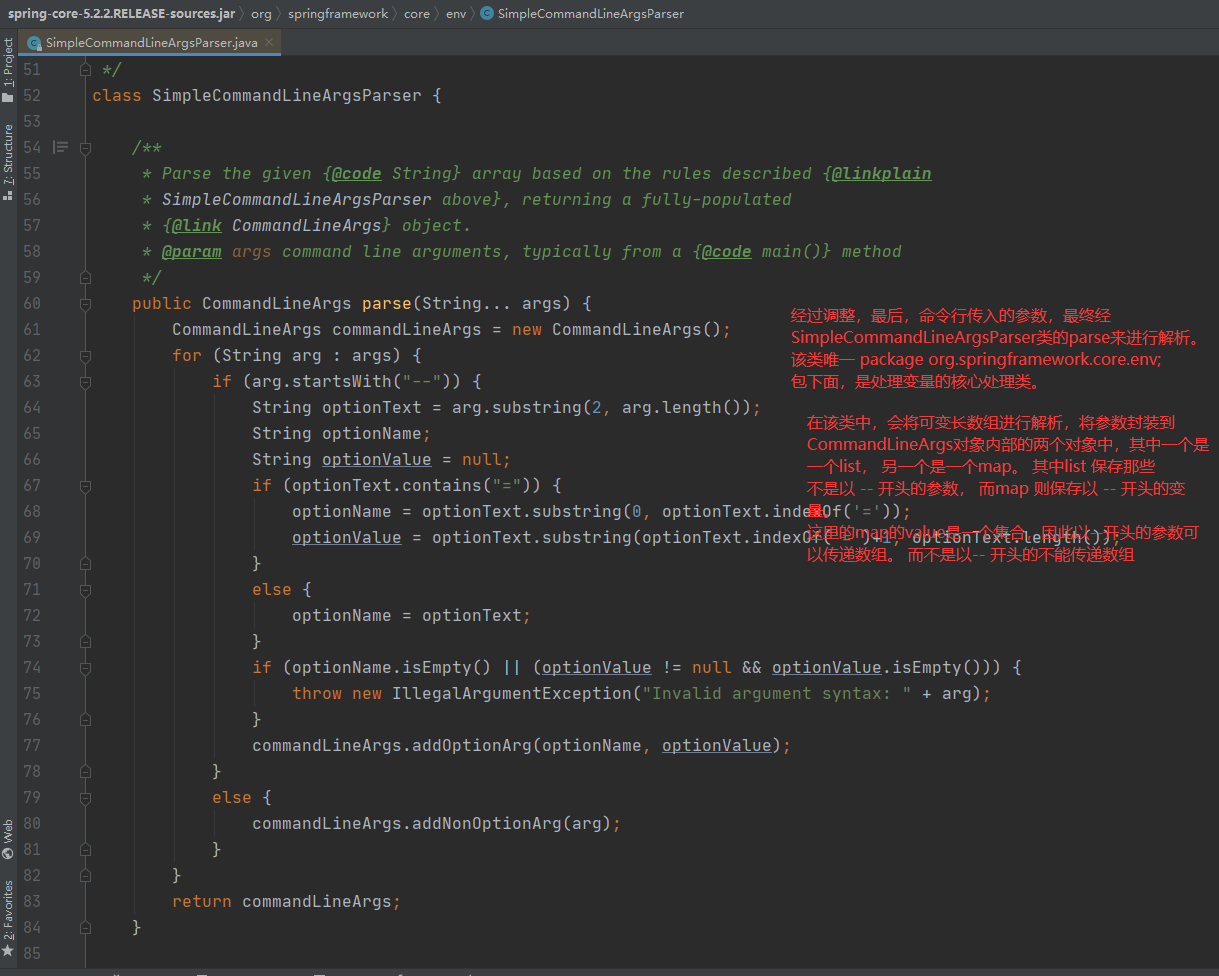


从这里可以看到将可变长参数args 传递给了两个对象，一个是args 变量，另一个是作为Source 类的参数来构造了一个Source 对象。

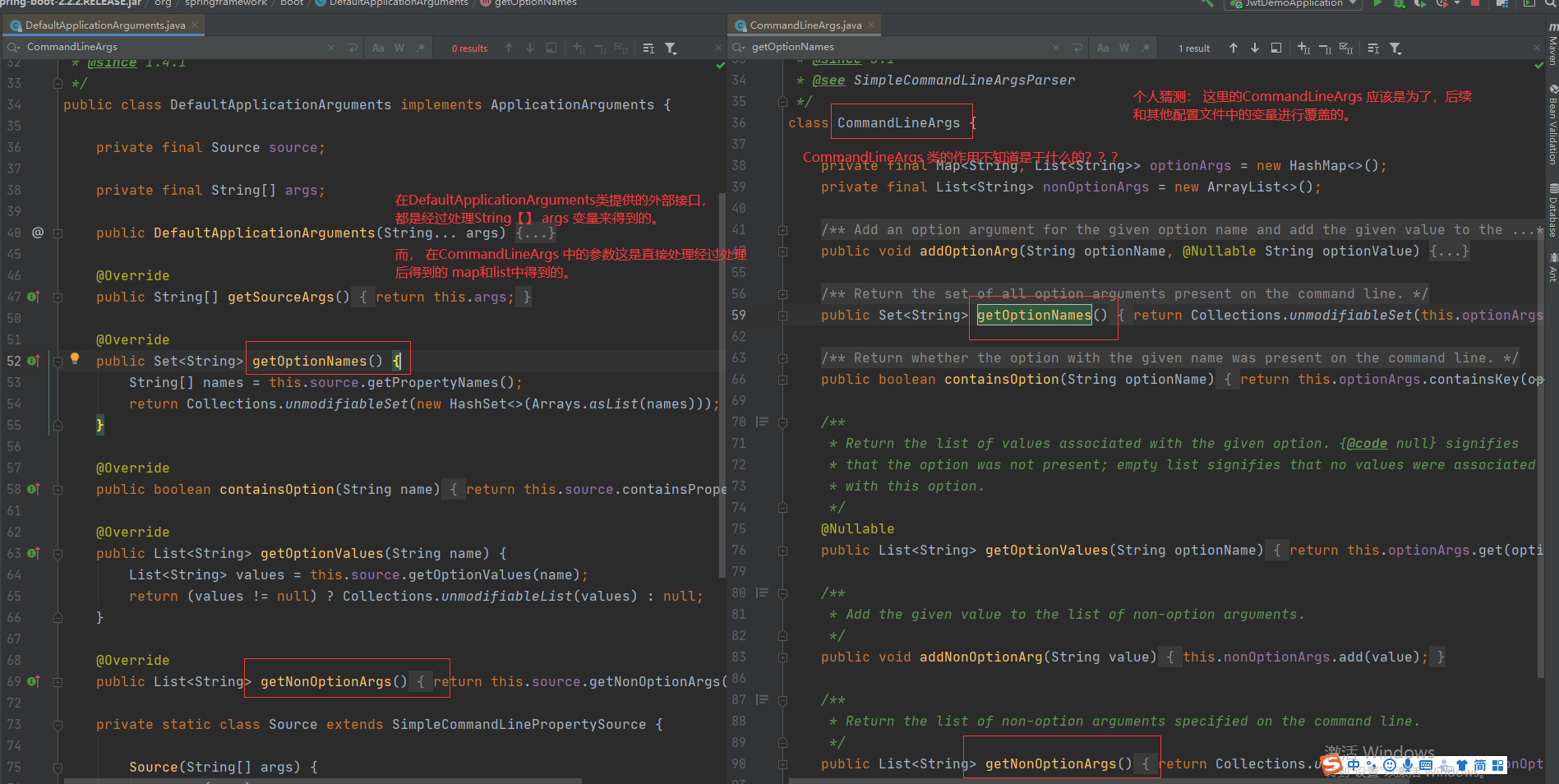
这里的Source 类是DefaultApplicationArguments 类的一个内部类。



经过深入，最后参数的处理是由SimpleCommandLineArgsParser 类的parse 方法进行解析，然后封装到CommandLineArgs 对象中。

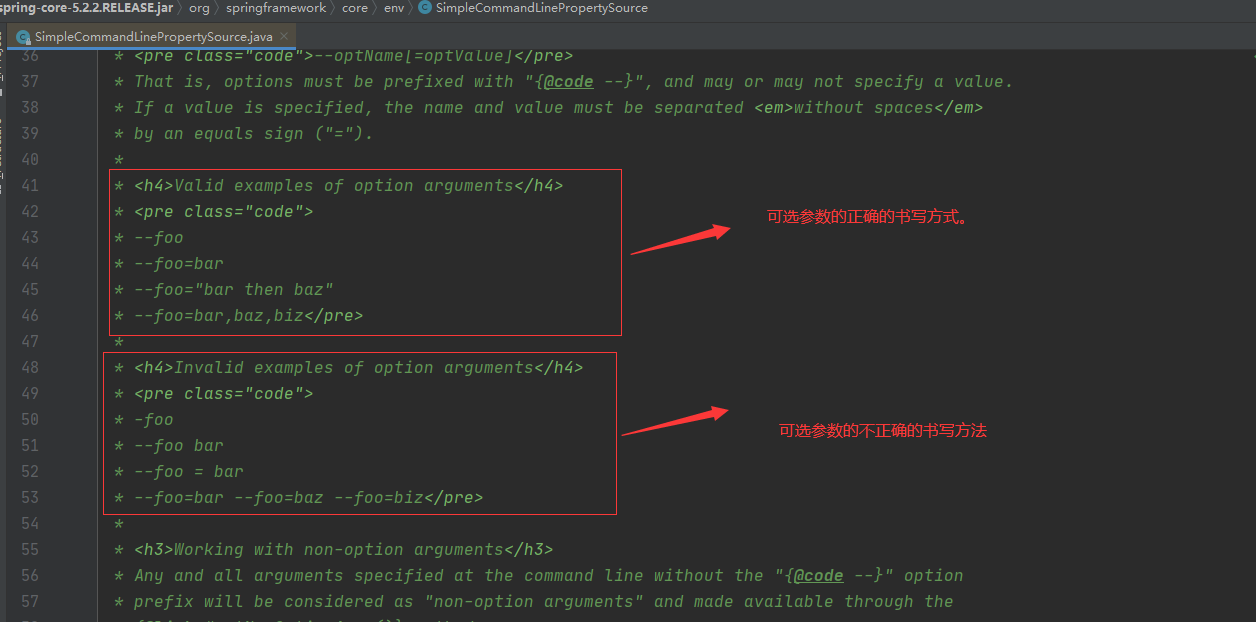




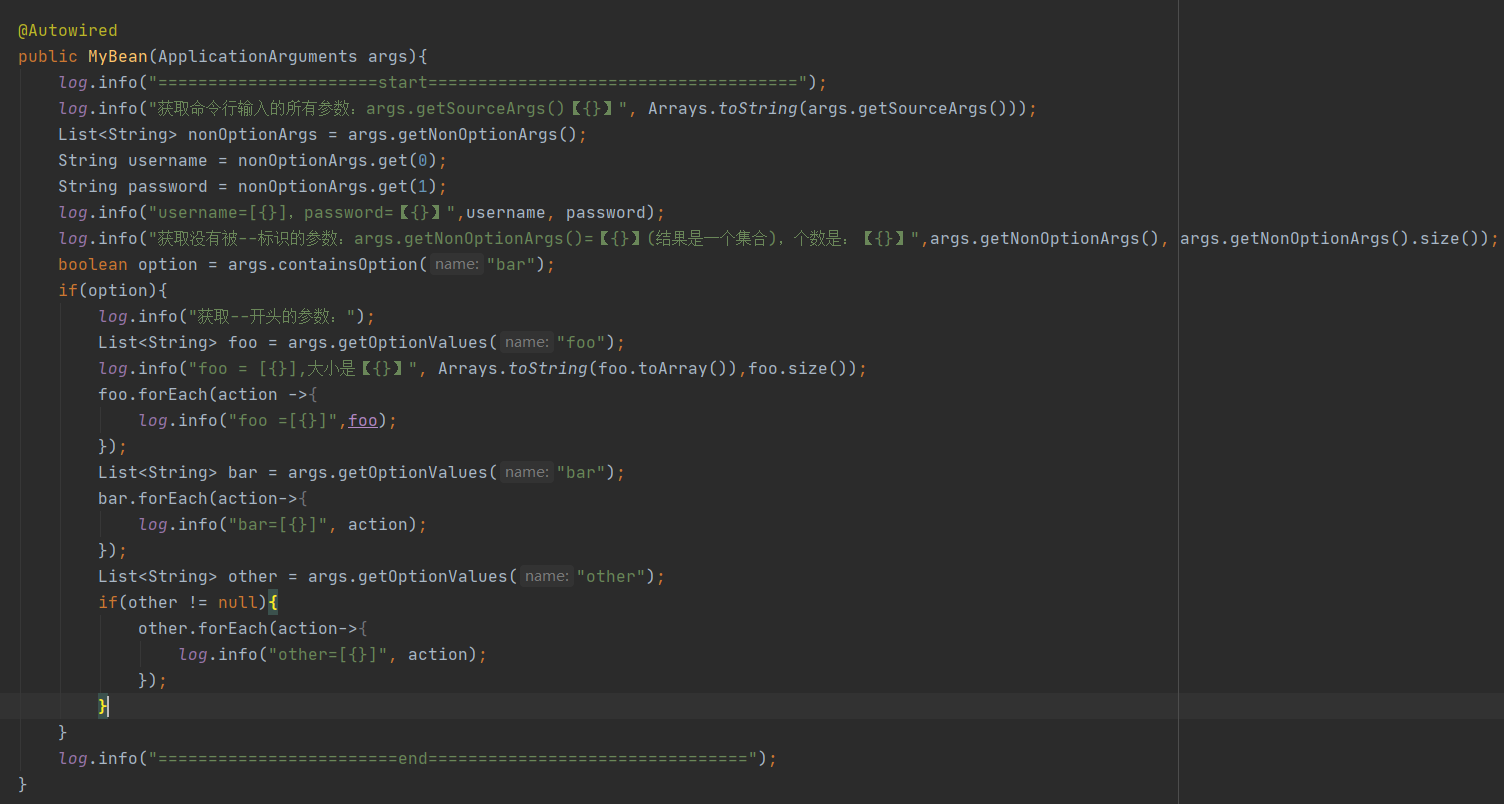


## Springboot中接收参数的正确书写方式

注意：书写的时候，不要有空格。



## Sprinboot 项目中获取参数的方式



这里args.getOptionValues(“key”) 得到的是一个集合。

注意：

这里目前没有发现比较好的直接获取username的方法。

## 疑惑

在springboot 中CommandLineArgs 类的作用不知道是干什么的？已经有了DefaultApplicationArguments 类，CommandLineArgs不知道干什么用。

解答： 这里的CommandLineArgs 就是对命令行参数的一个抽象。 这里的CommandLineArgs类就代表了命令行参数，而且将命令行参数分为可选参数和非可选参数

