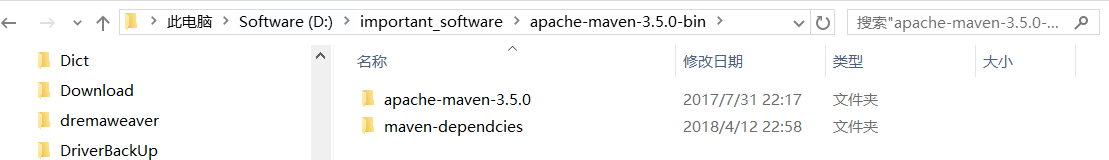
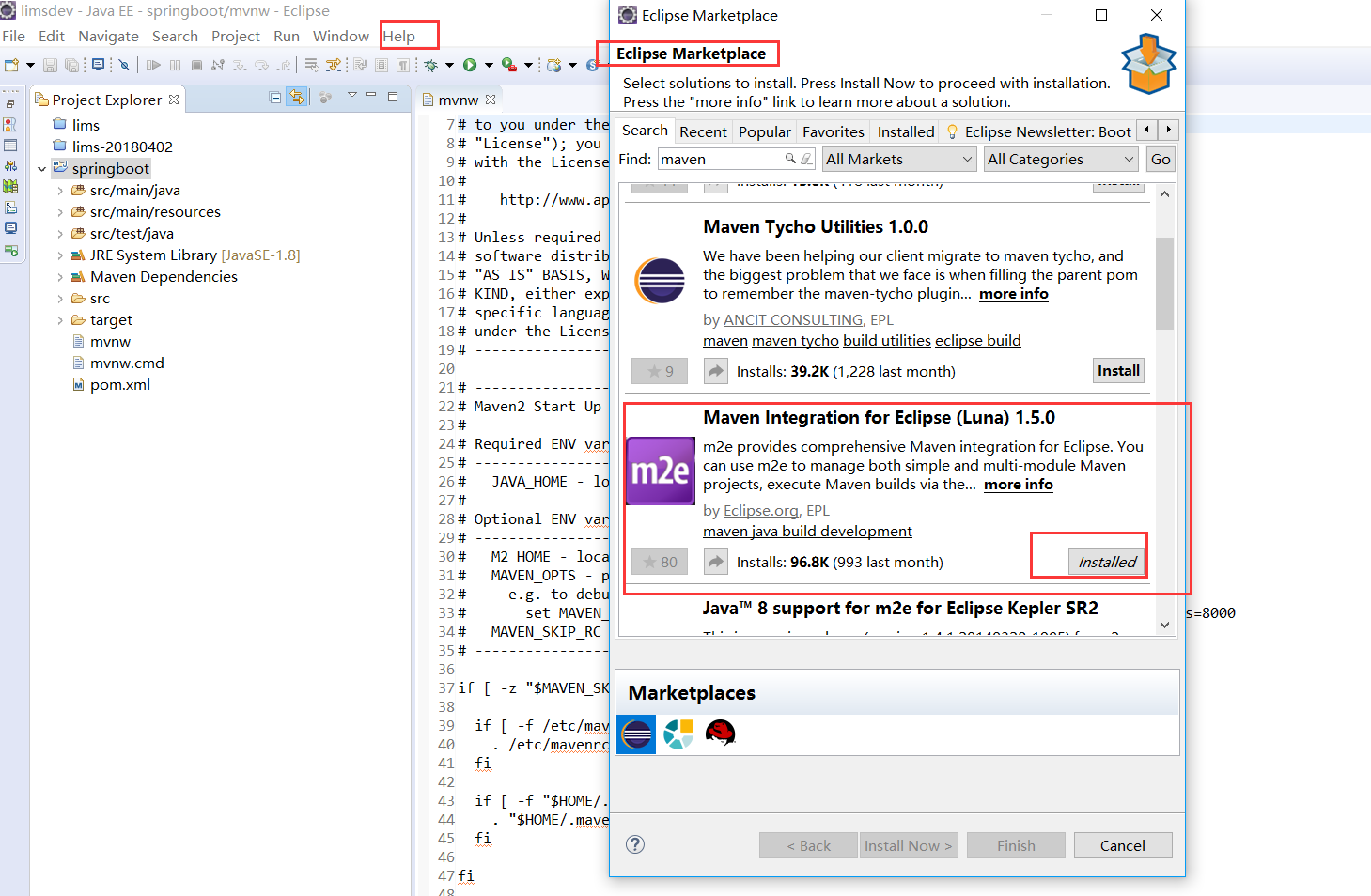
Spring boot 随手记

#### 使用Eclipse，集成maven

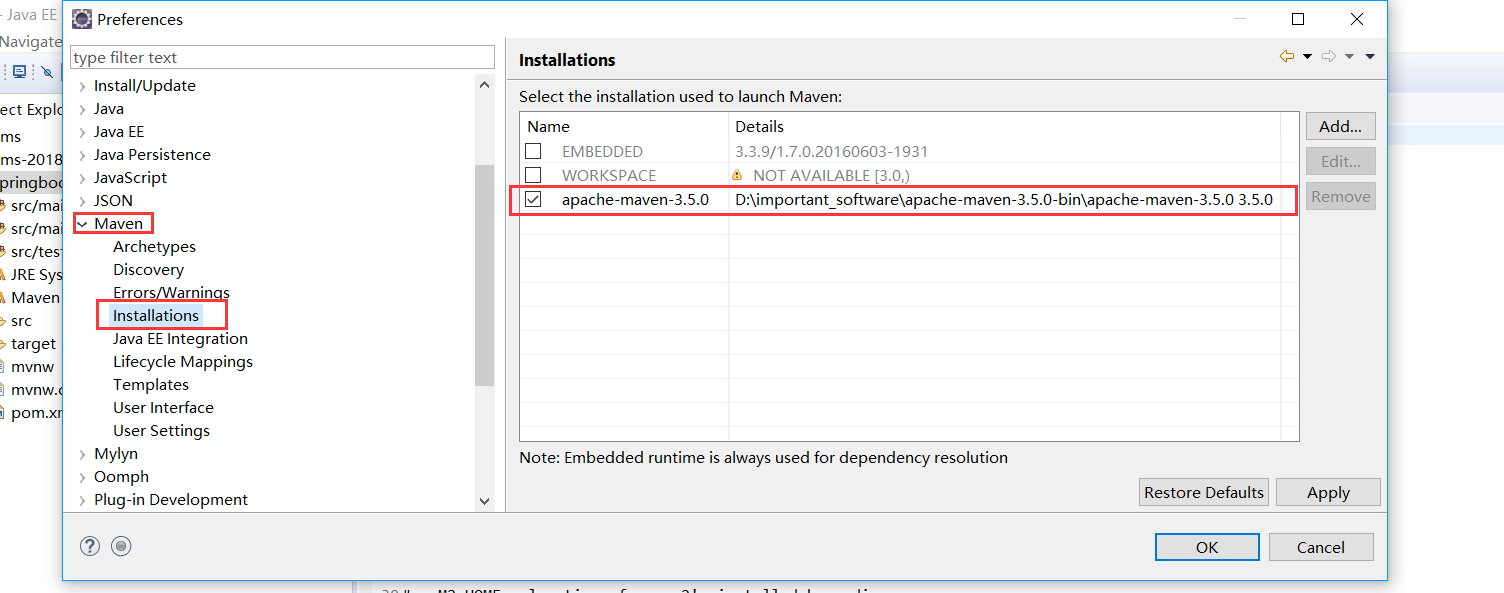
下载解压maven包

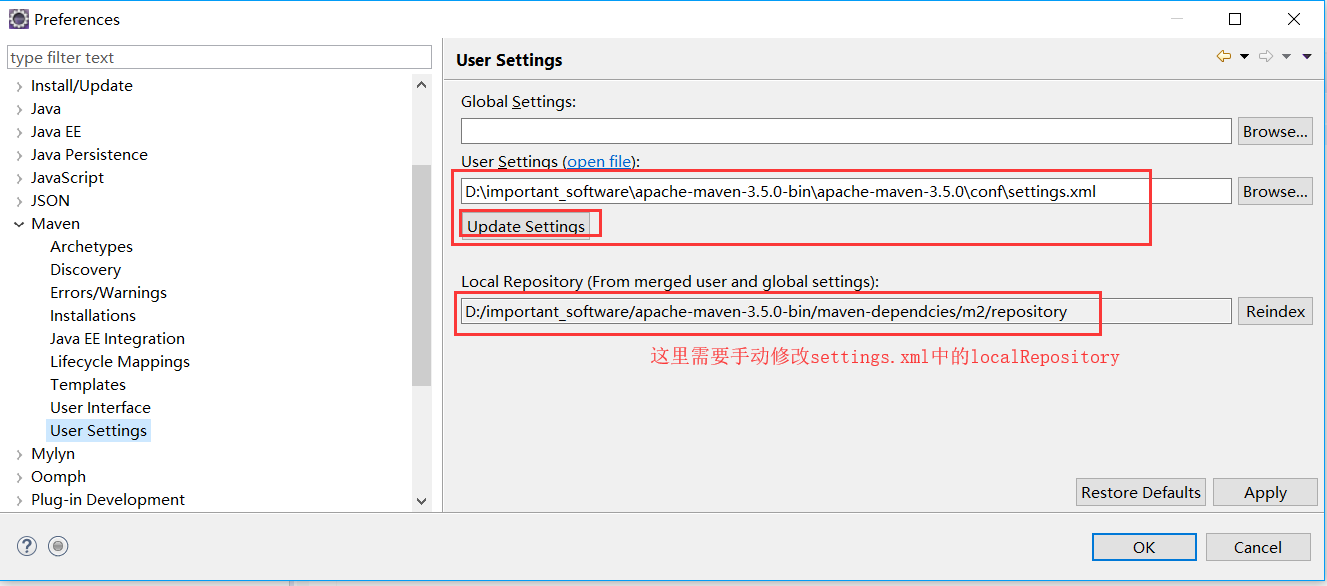


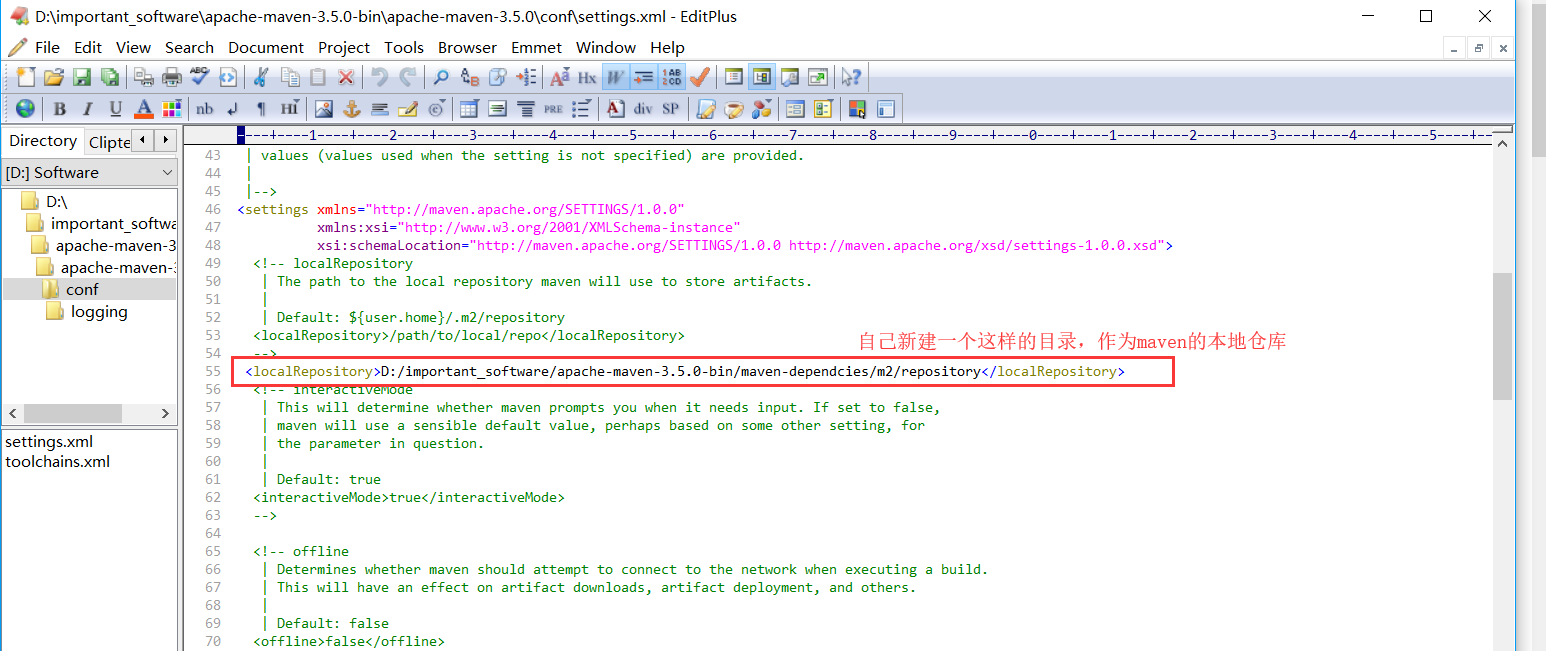
安装maven插件



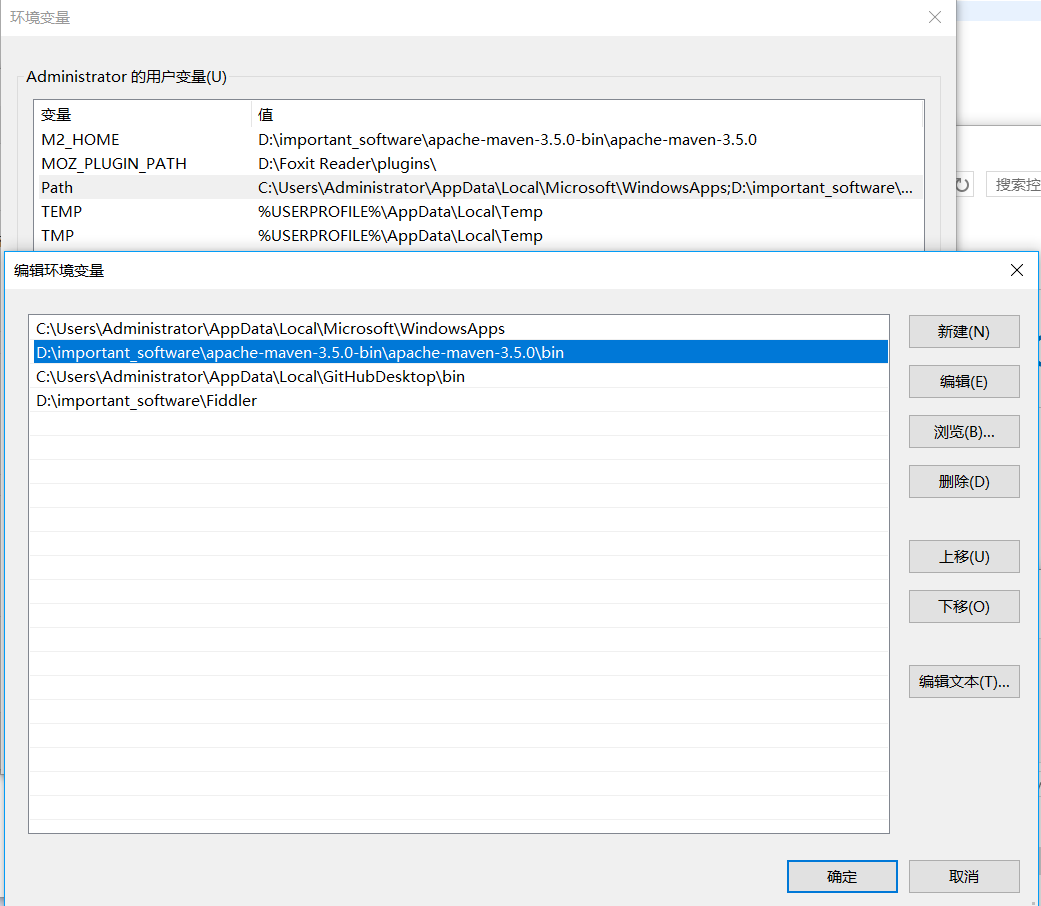
关联包



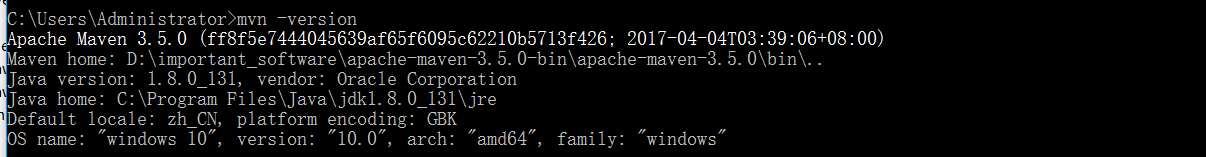




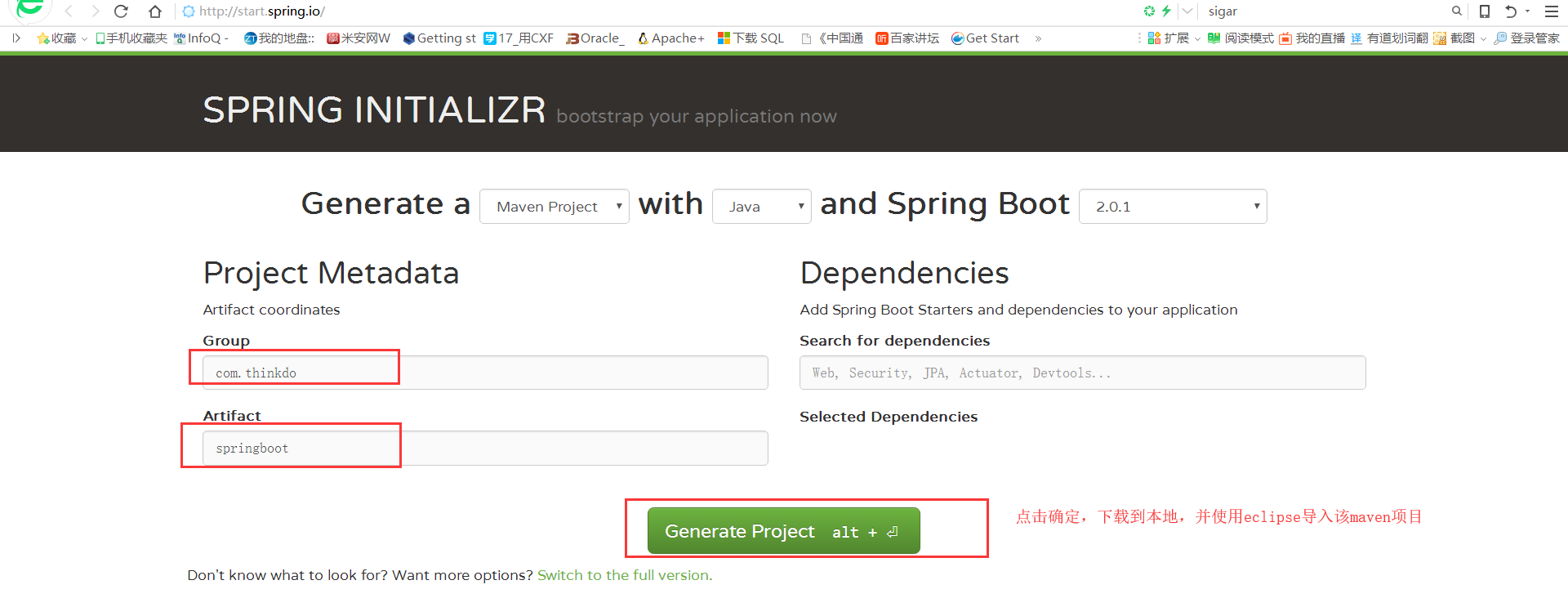
加入环境变量

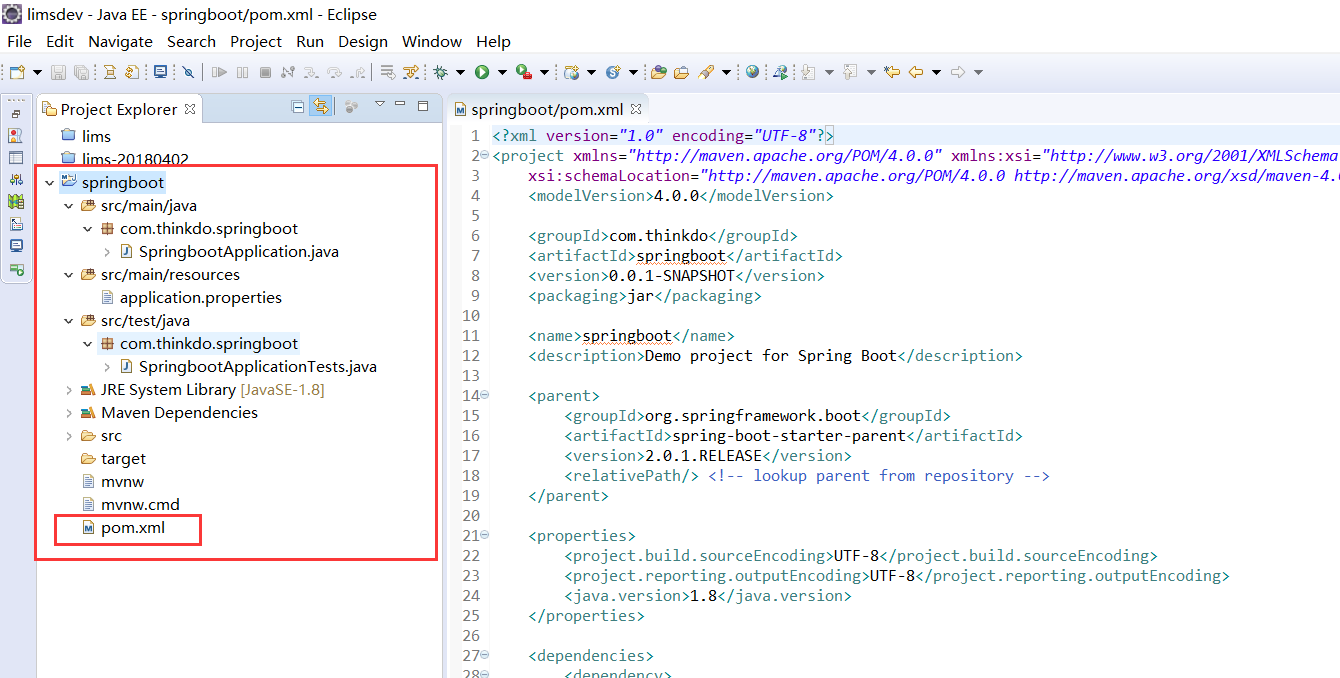


验证是否加入到环境变量中了：

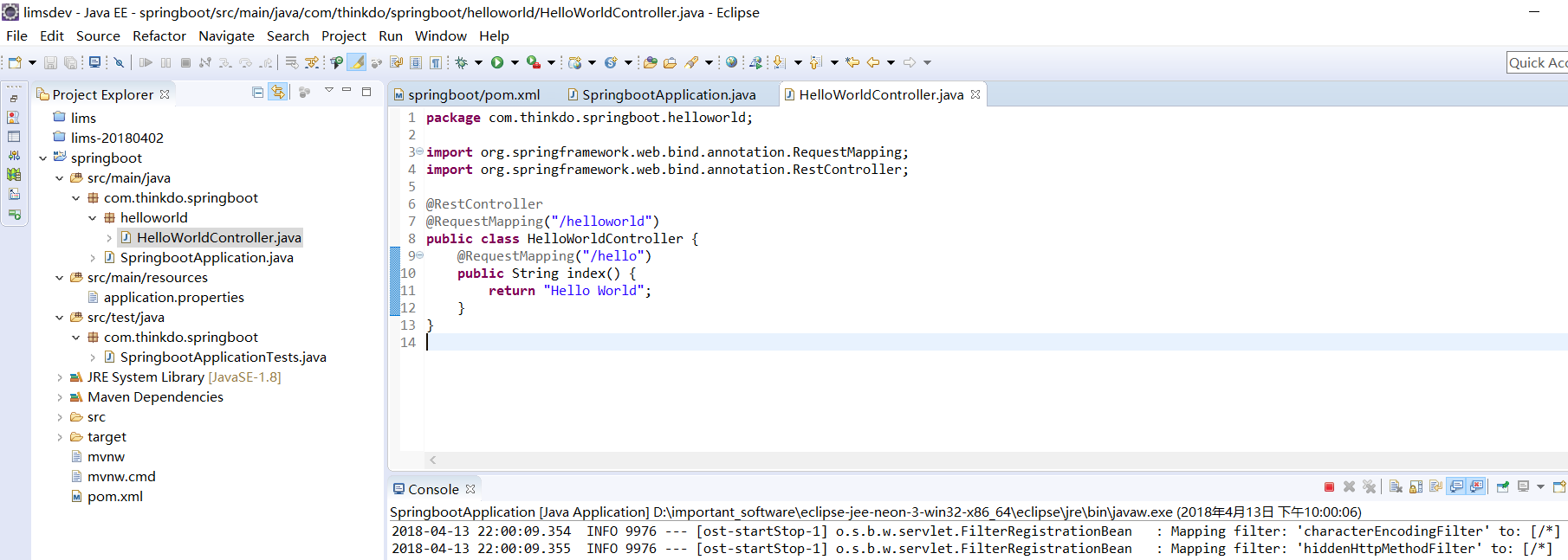


#### 跑一个spring boot demo起来

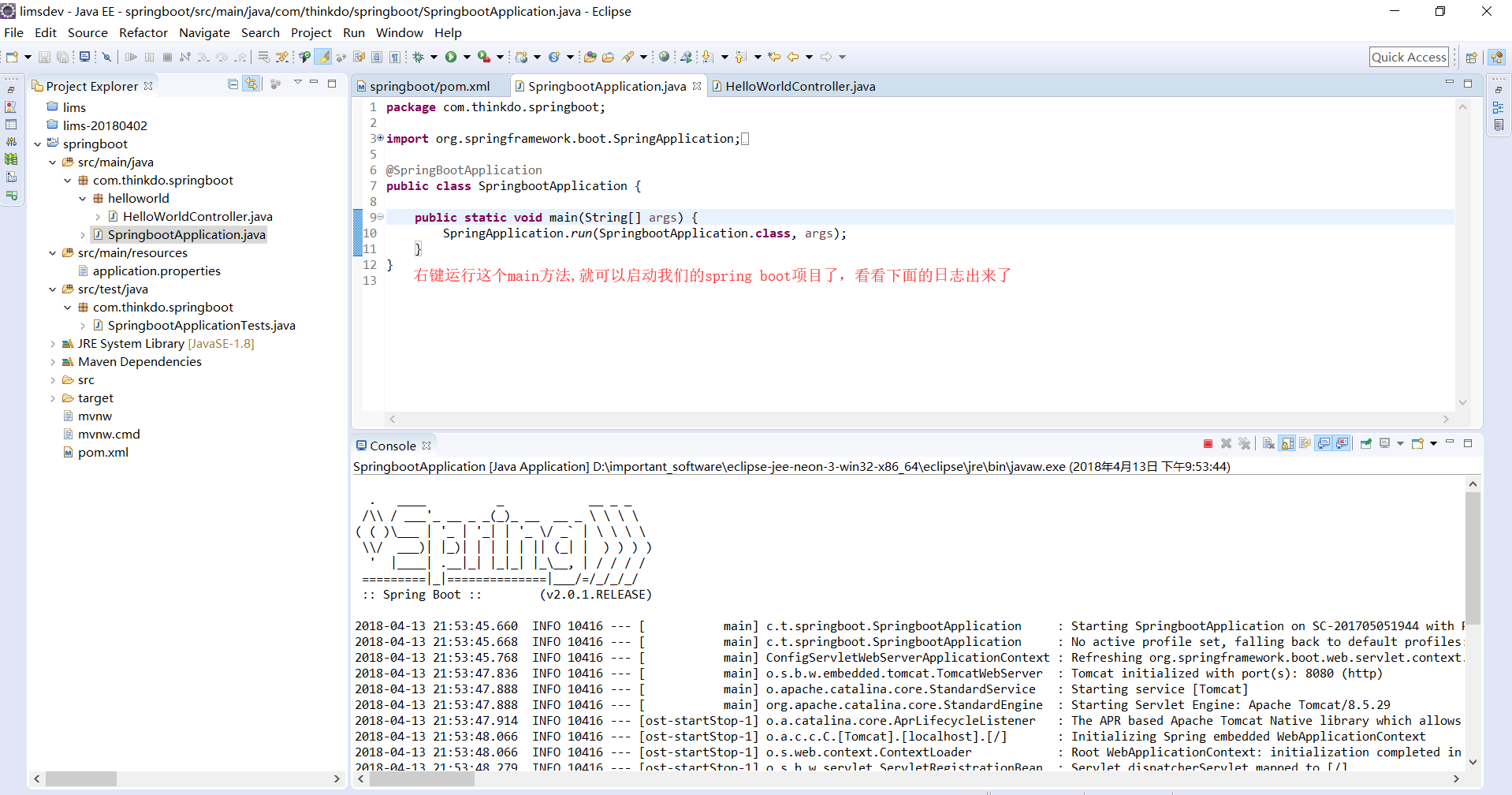




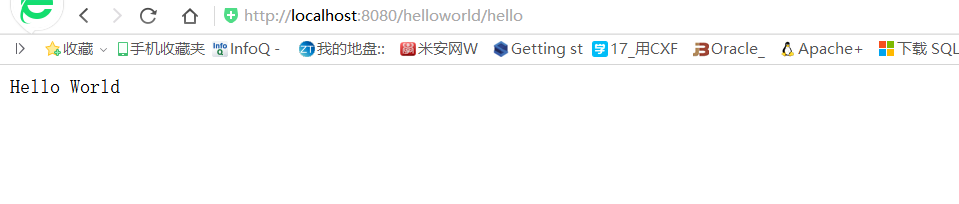
添加一个Controller类：



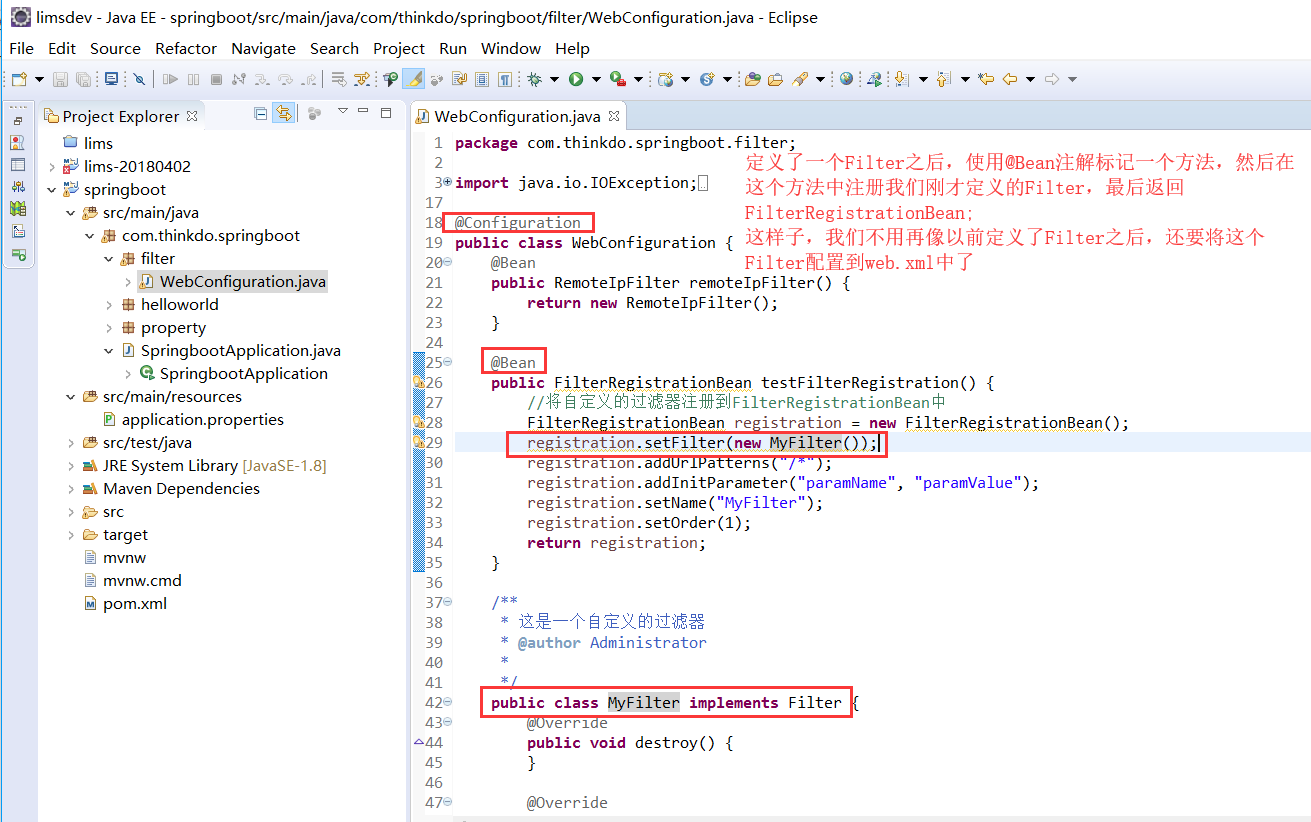
启动项目



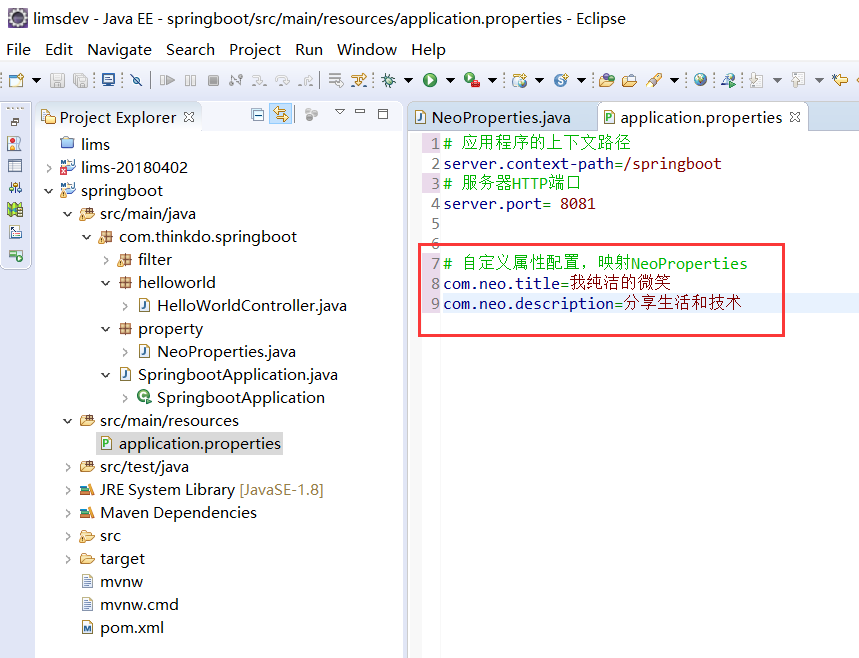
结果

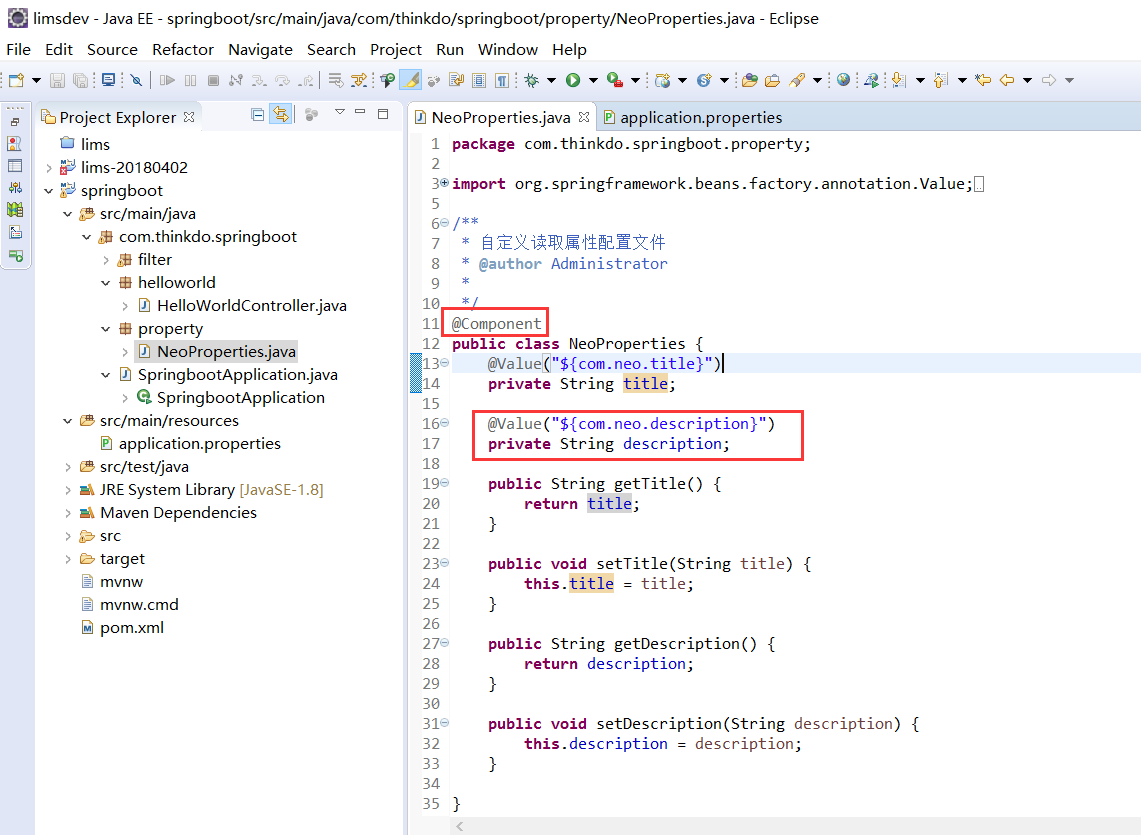


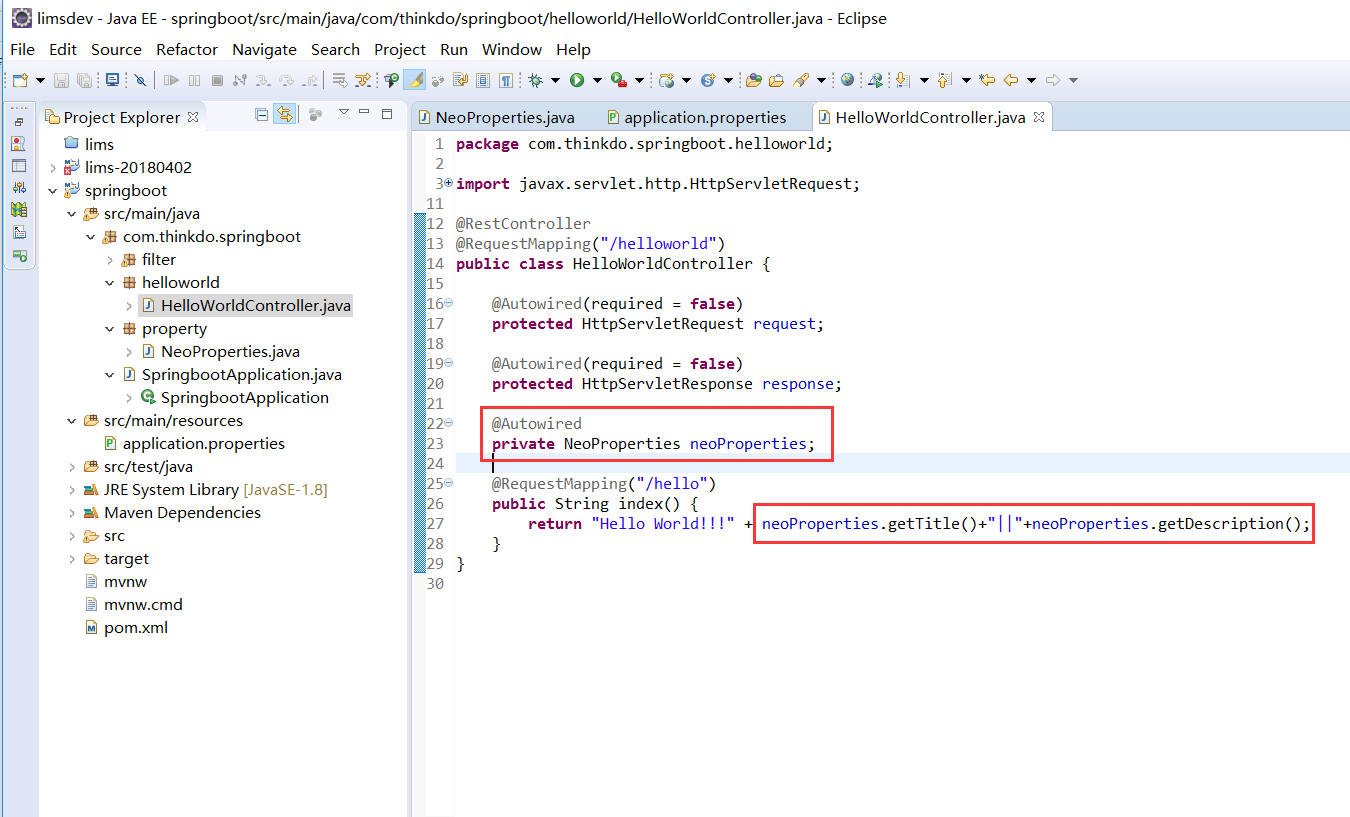
#### 添加过滤器Filter



#### 读取在application.properties中配置的值







这个过程中我遇到了读取配置属性中文乱码的问题，我使用的是eclipse环境，解决办法是：

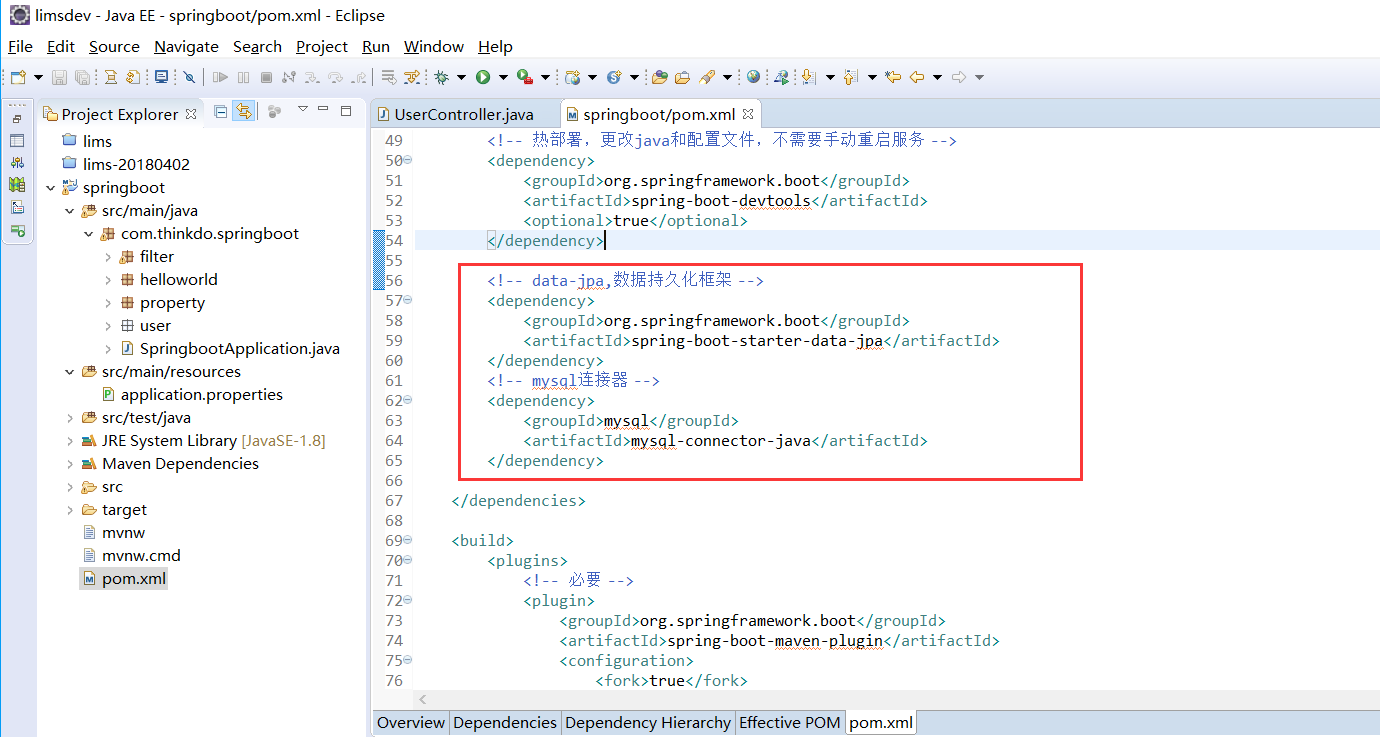
Eclipse中安装一个插件PropertiesEditor来重新编辑properties文件，在线安装地址：

http://propedit.sourceforge.jp/eclipse/updates/

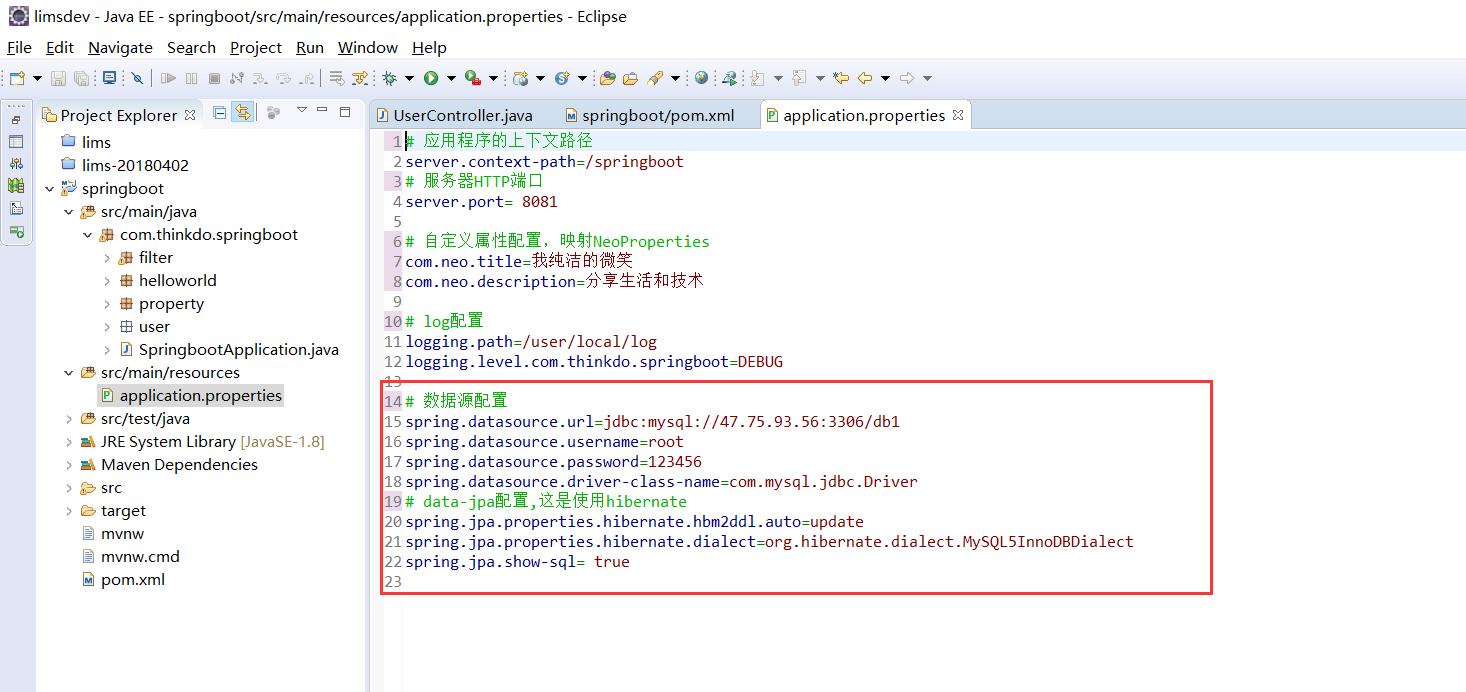
#### data jpa

这个在使用中碰到最多的问题是maven 仓库中下载下来的依赖包有时候有问题，需要进入到有问题的包下，将其全部删除，然后maven--update project

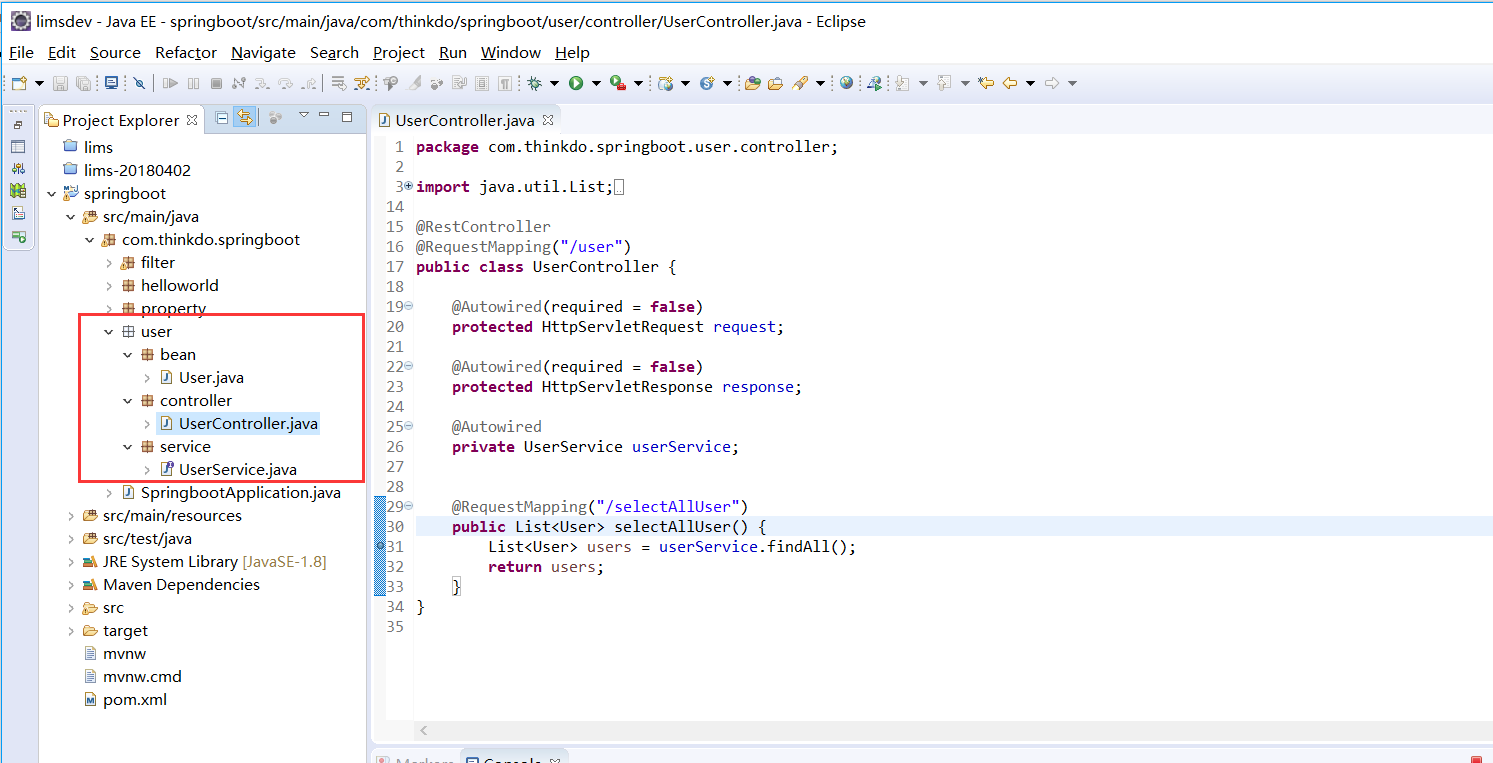
在pom.xml中配置依赖



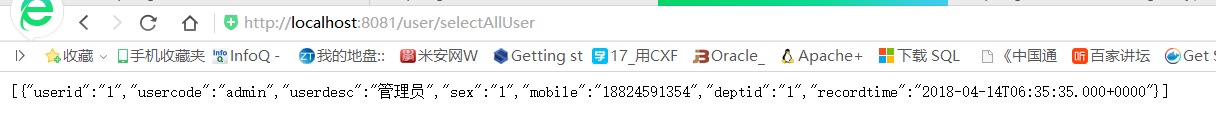
在application.properties中配置数据源和jpa



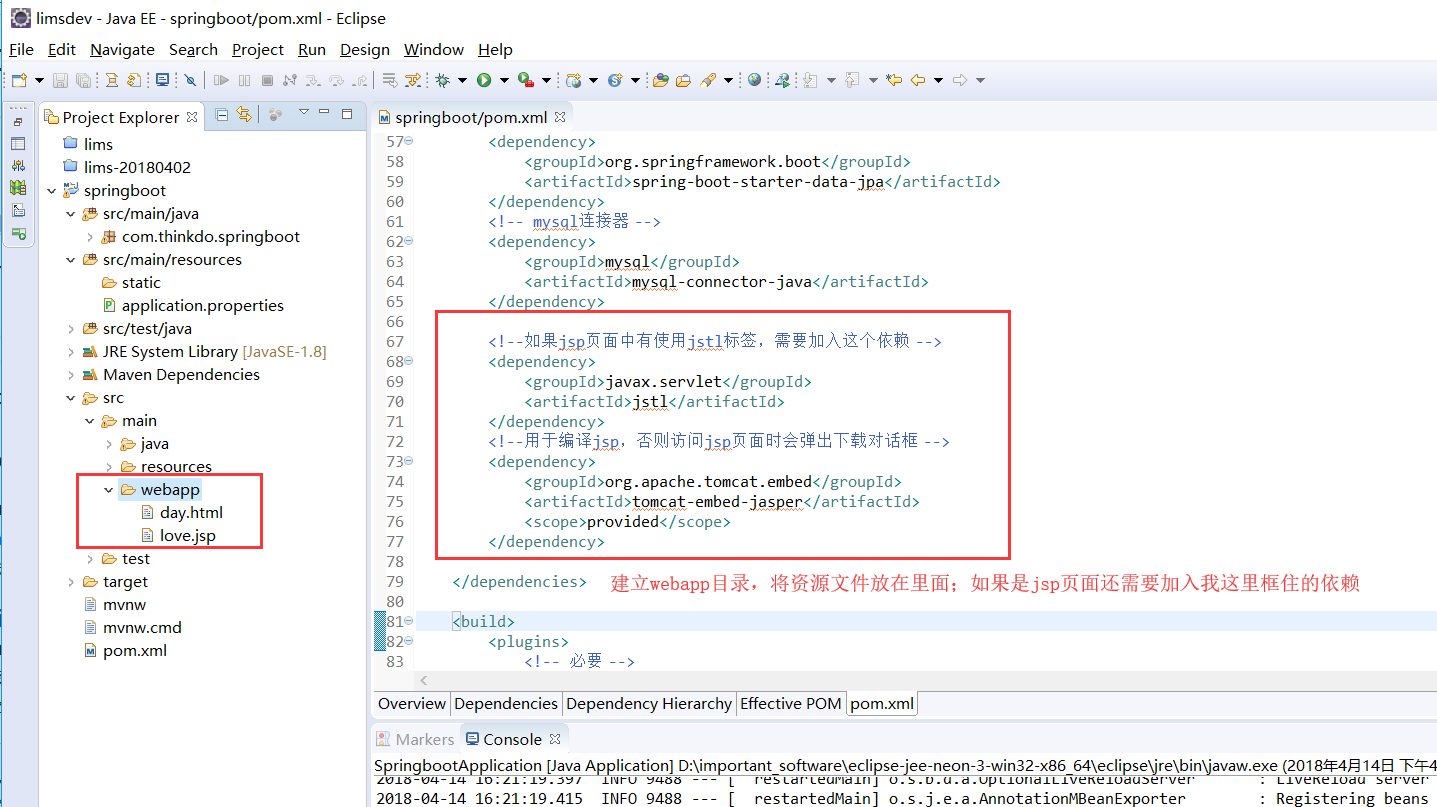
建立user模块



其中对实体对象需要加注解@Entity,@Table(name="base\_user"),里面的表名不能写错，否则项目运行起来会自动创建不能存在的表；@Id、@Column注解在实体对象的成员属性上，同样不能注入错误，字段要一一对应，数据库中没有在实体对象中有的成员需要加@Transient



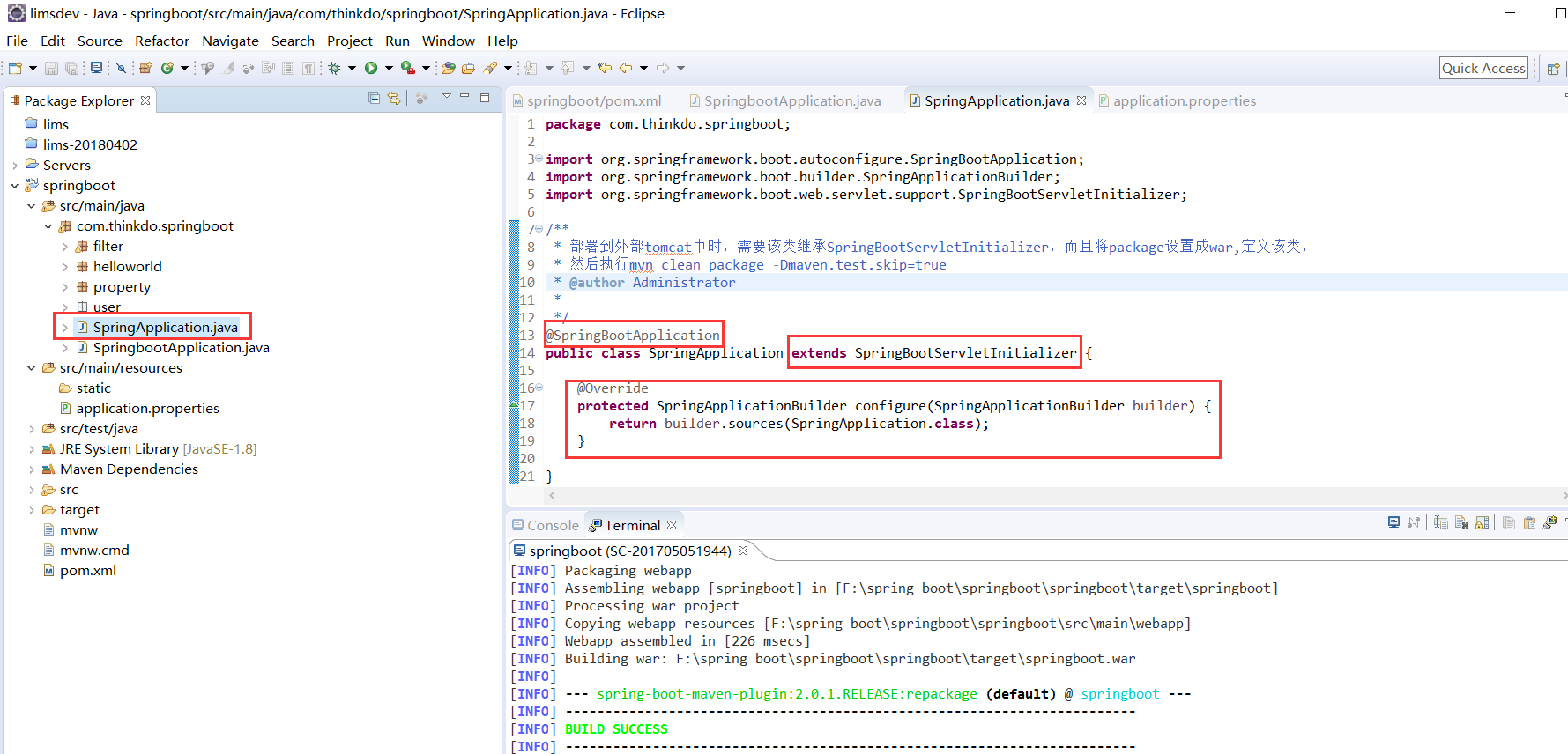
#### springboot中访问jsp页面





#### springboot项目部署到外部tomcat容器

需要修改我们之前定义好的SpringbootApplication如下：



修改pom.xml，打包方式为war，对内置tomcat依赖域设置provided

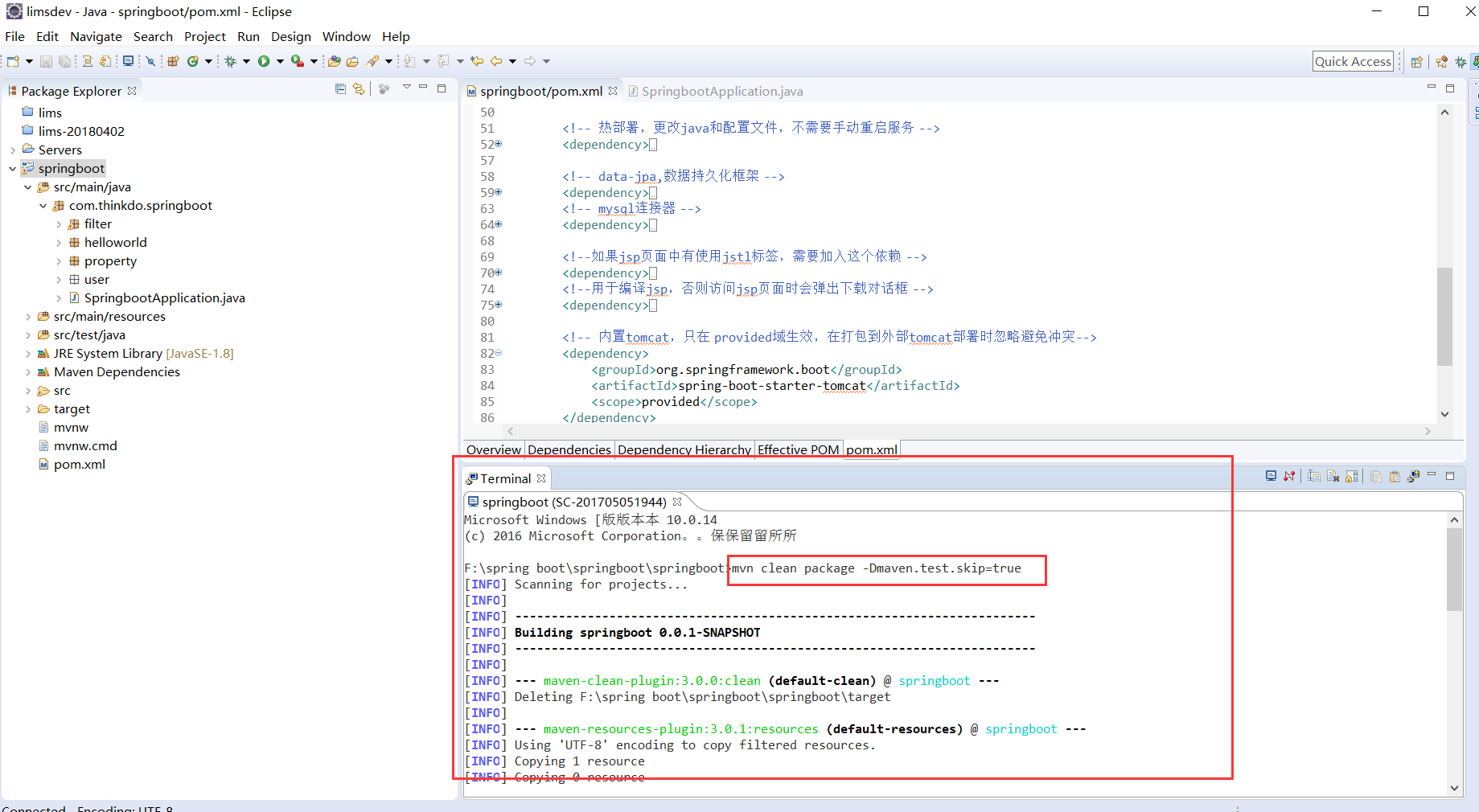


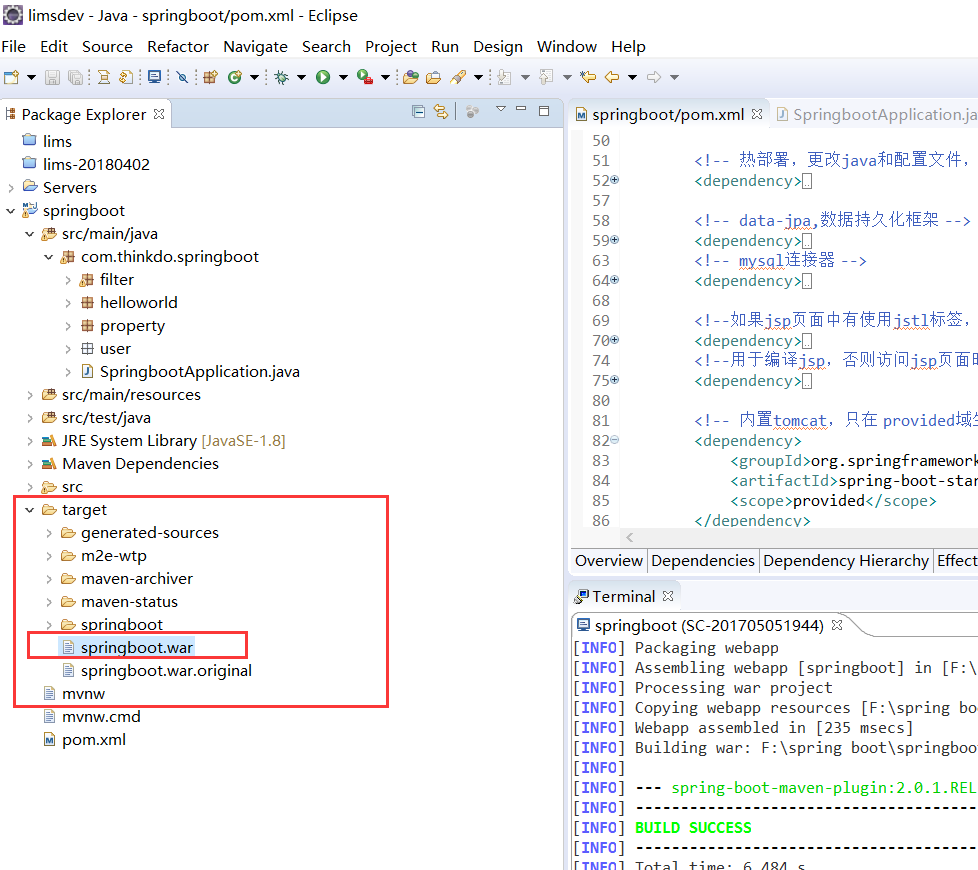


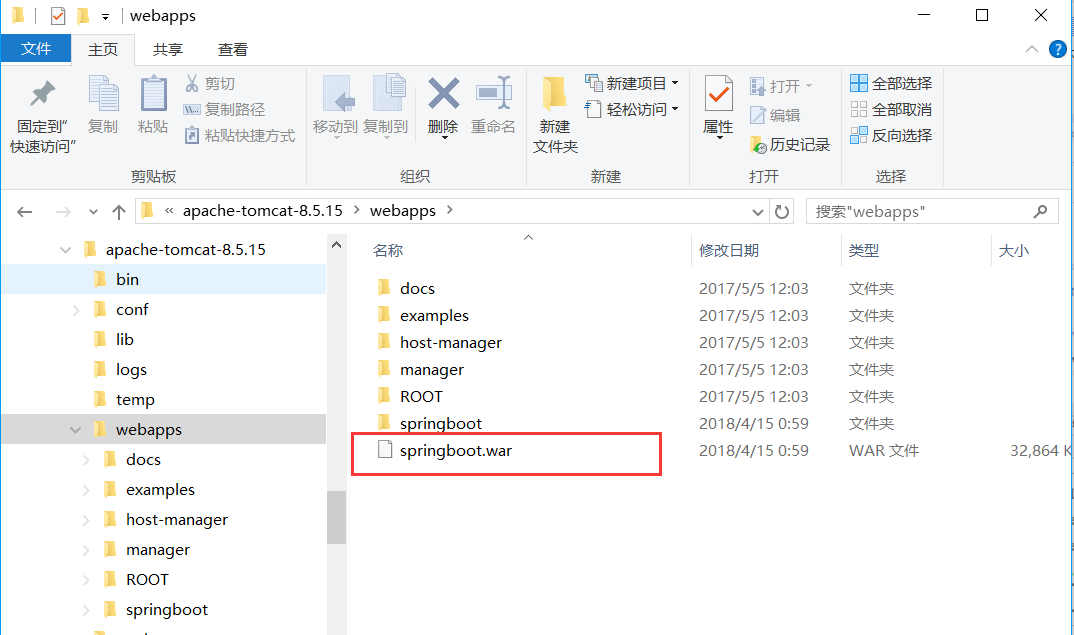
然后右键项目 ，show in terminal 出现本地dos命令窗口

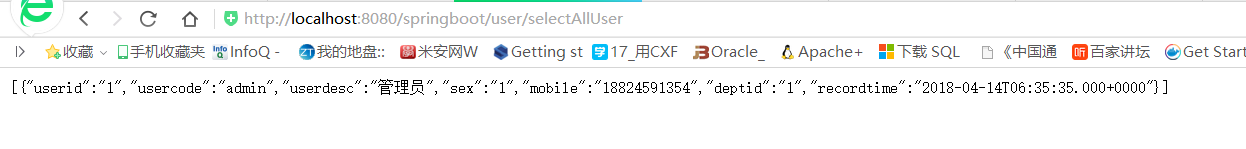
输入mvn clean package -Dmaven.test.skip=true

执行成功后会在target目录下生成war包，将war包放到外置tomcat的webapps下面，然后启动外置tomcat就可以了。









#### 使用redis