# 实验2-1 Spring IoC

一．实验目的

1．掌握Spring IoC工作原理

2．掌握Spring项目的创建、配置、注入和测试

3．掌握Android App注册/登录界面设计

4．会使用在线文档：

Spring官网：<https://spring.io/projects/spring-boot>

Apache Maven官网：<https://maven.apache.org/index.html>

MyBatics官网：<https://mybatis.org/mybatis-3/zh/index.html>

Maven仓库：<https://mvnrepository.com/>

Spring Boot API：<https://docs.spring.io/spring-boot/docs/current/api/>

Android API：<https://developer.android.google.cn/reference?hl=en>

二．实验环境

计算机，网络，Windows 64，Jdk 8+，Intellij IDEA，Android Studio，MySQL，Microsoft Office

三．实验内容

任务1：创建Spring项目

任务2：实现Spring的IoC

任务3：Spring注入

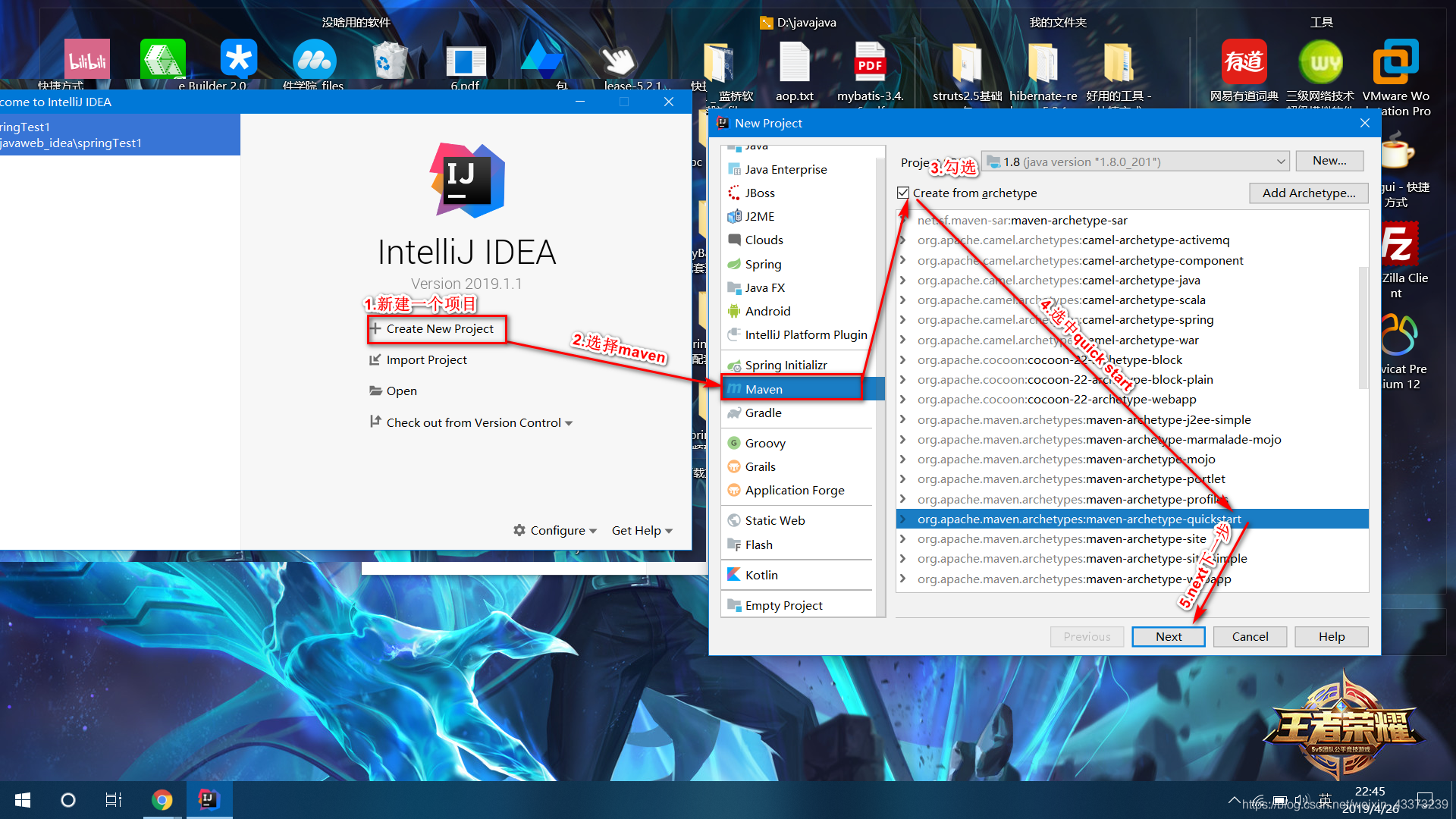
四．实验步骤

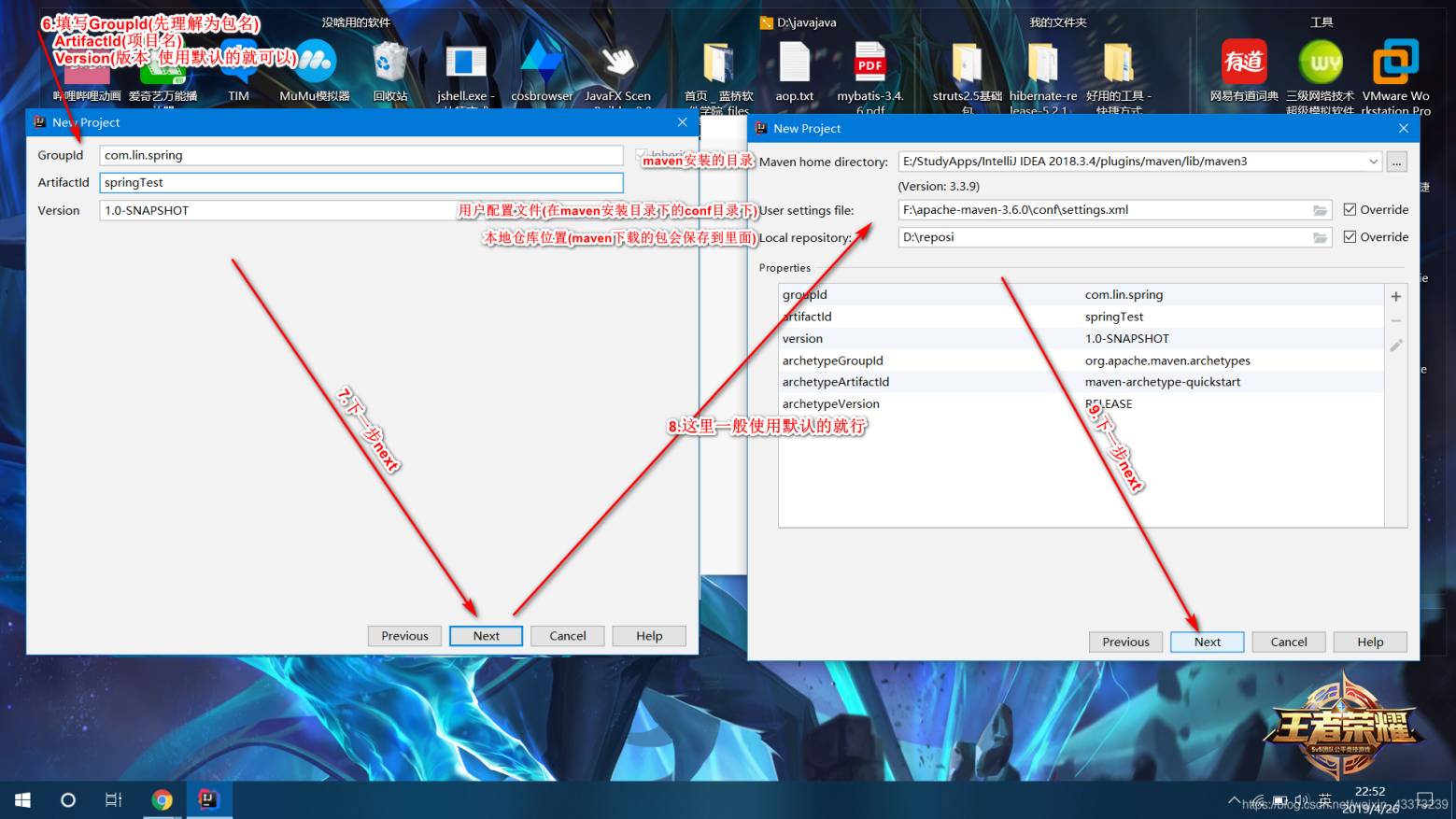
任务1：创建Spring项目

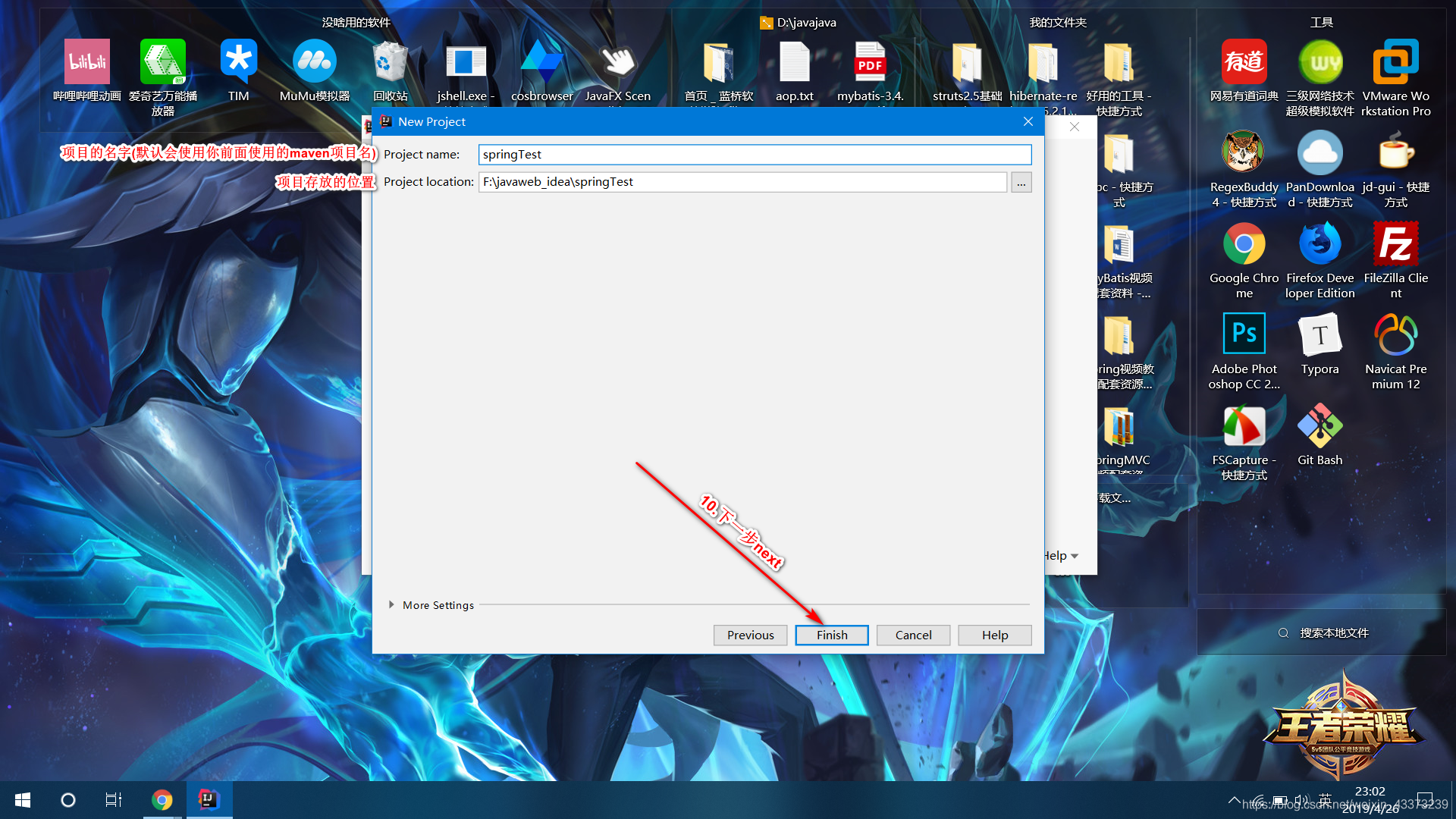
任务说明：用IDEA创建一个基于Spring Framework的程序，包名：com.binger ，项目名：001-spring

具体步骤(因Idea的版本不同，以下截图仅供参考)：

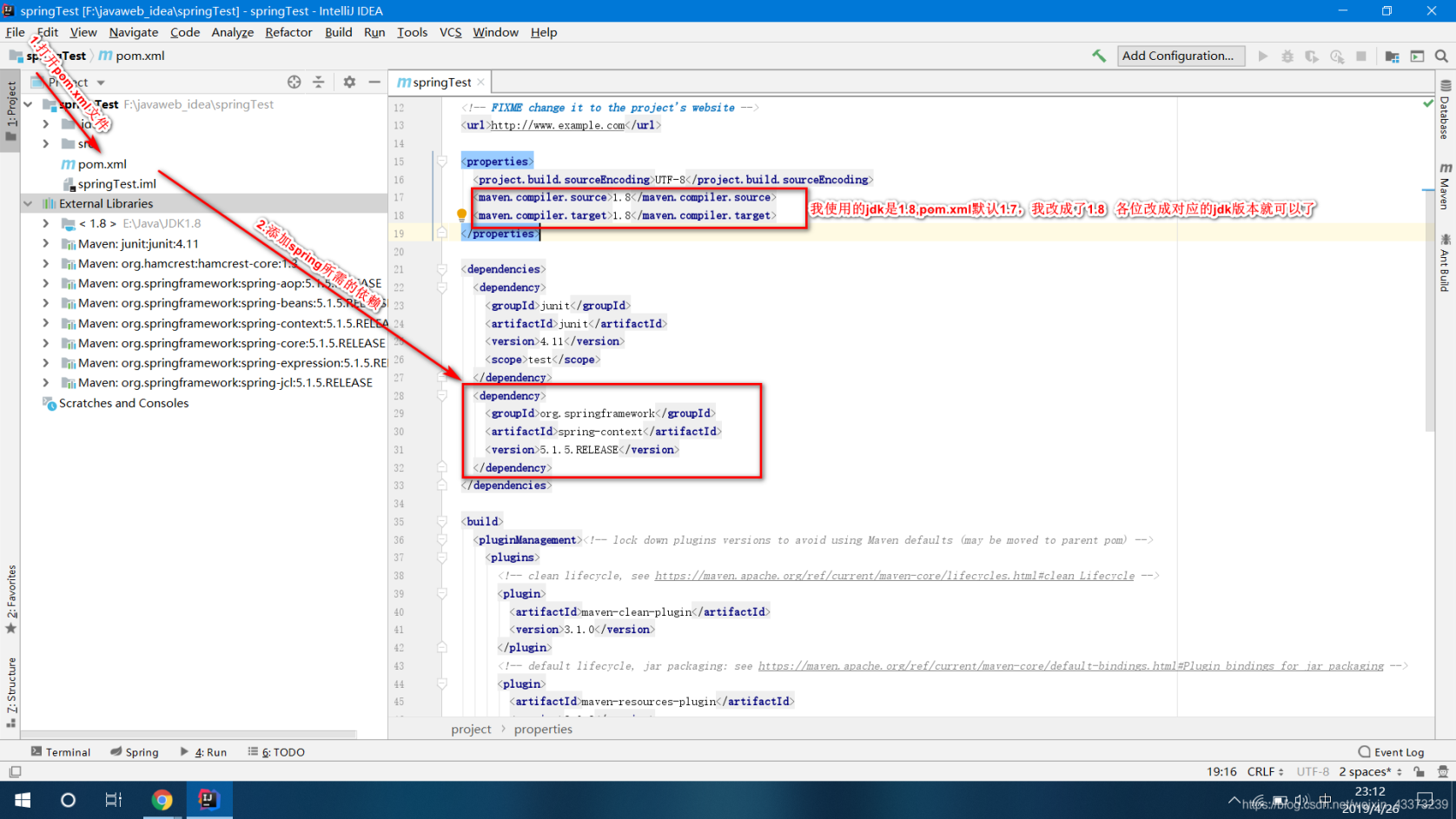
1.打开idea，新建项目。



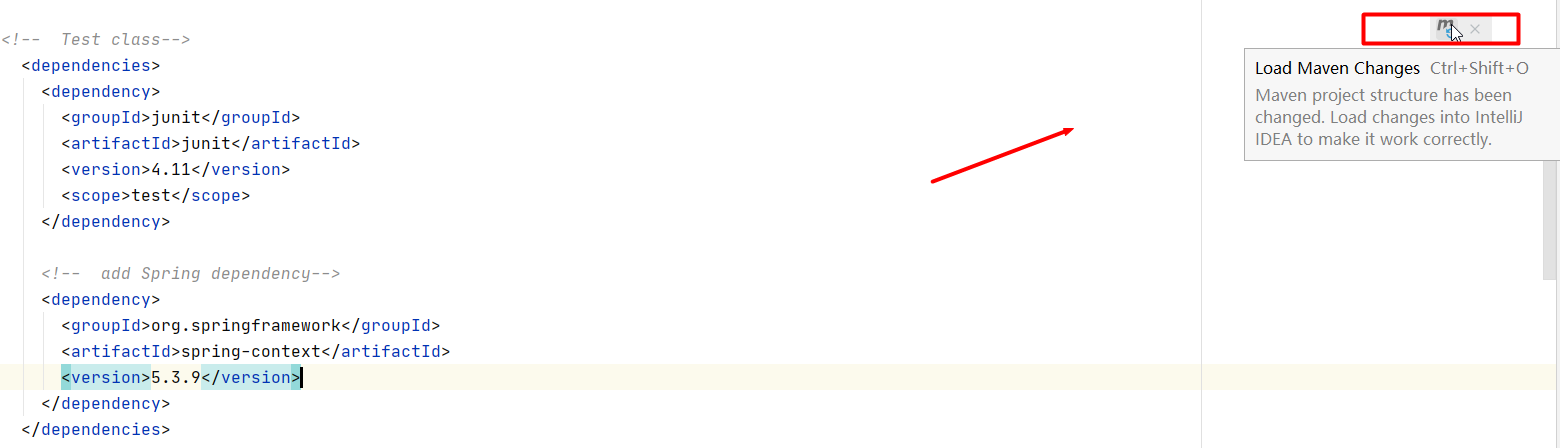




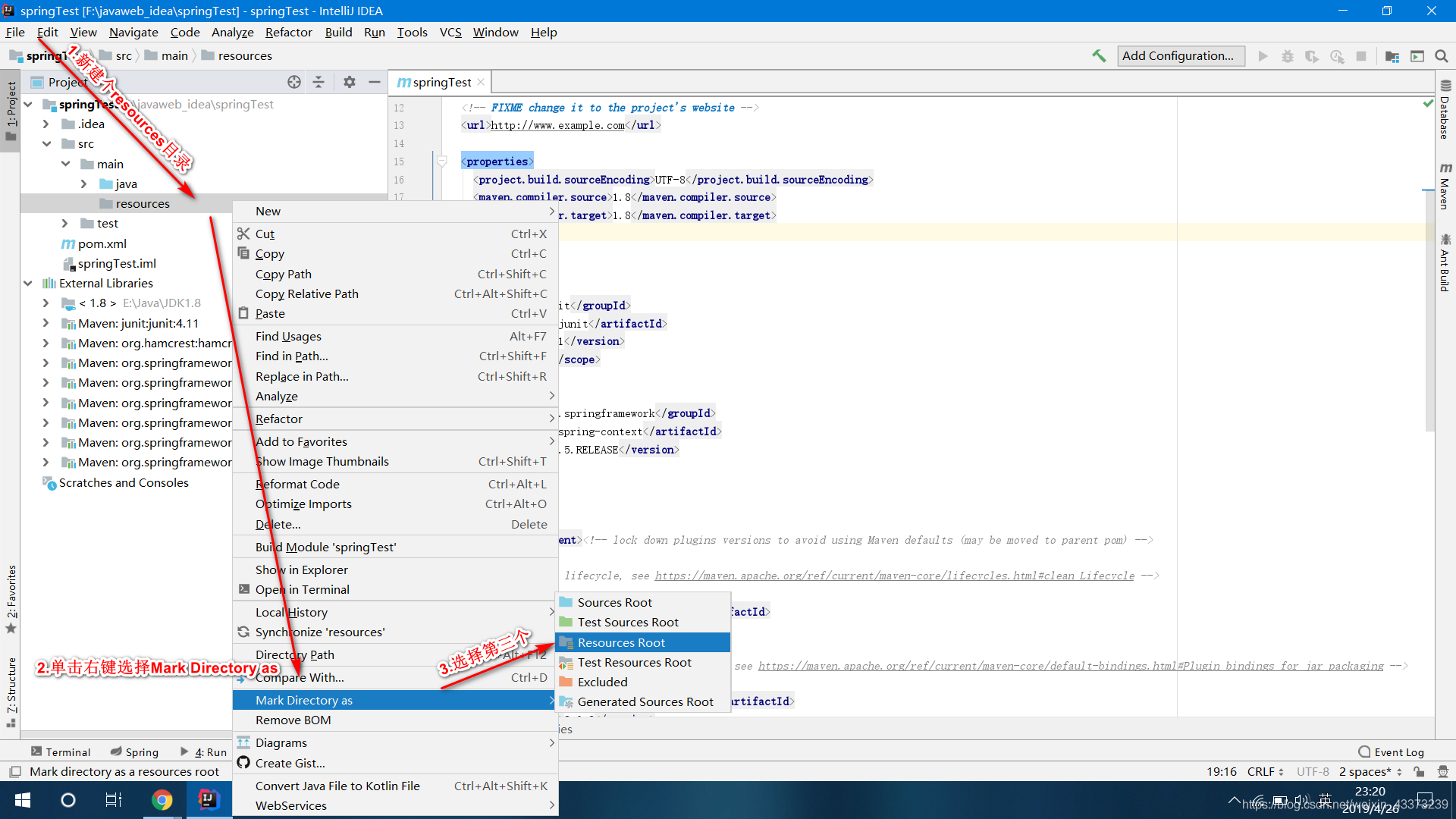
2.修改pom.xml文件，添加spring所需的依赖。



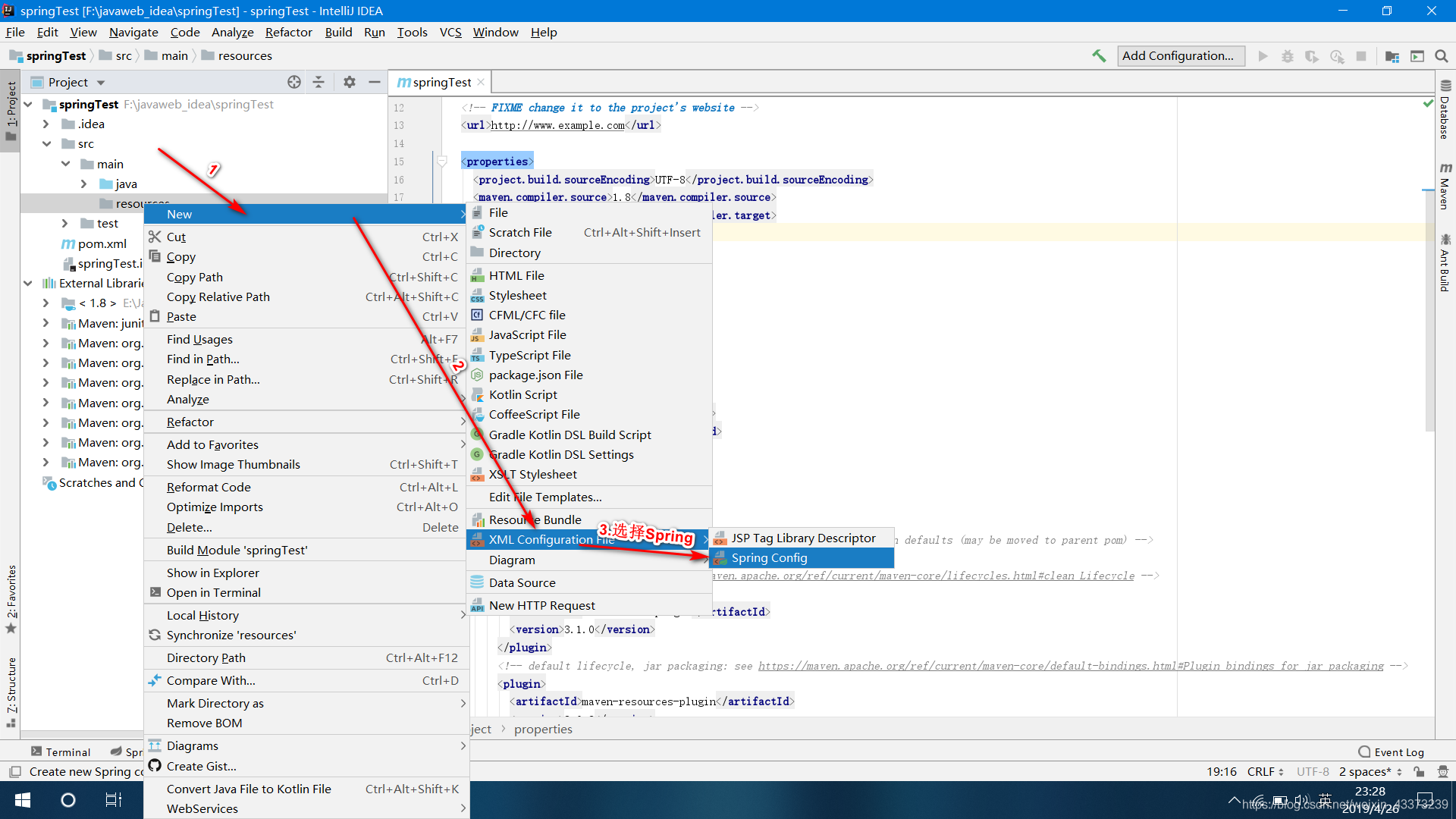
添加后import一下就可以了



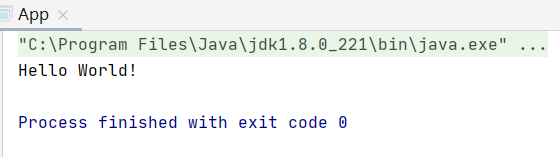
3.在main下新建一个resources（也可以取别的名字）目录，并设置为资源根目录，这个资源根文件就是class存放的位置（就把他当classs文件夹吧）



4.在resources目录下新建一个applicationContext.xml文件(如果spring的依赖导入成功了idea可以快捷新建)



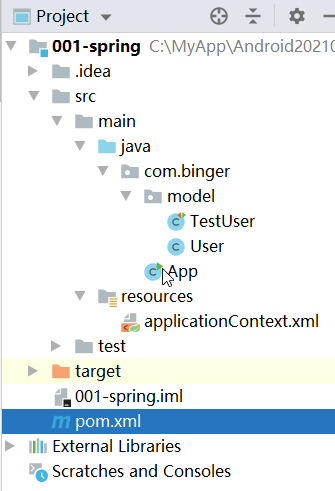
5.运行效果



任务2：实现Spring的IoC

任务说明：本任务接着任务1继续做实验，实现Spring的IoC，并从控制台输入用户信息。

项目结构：

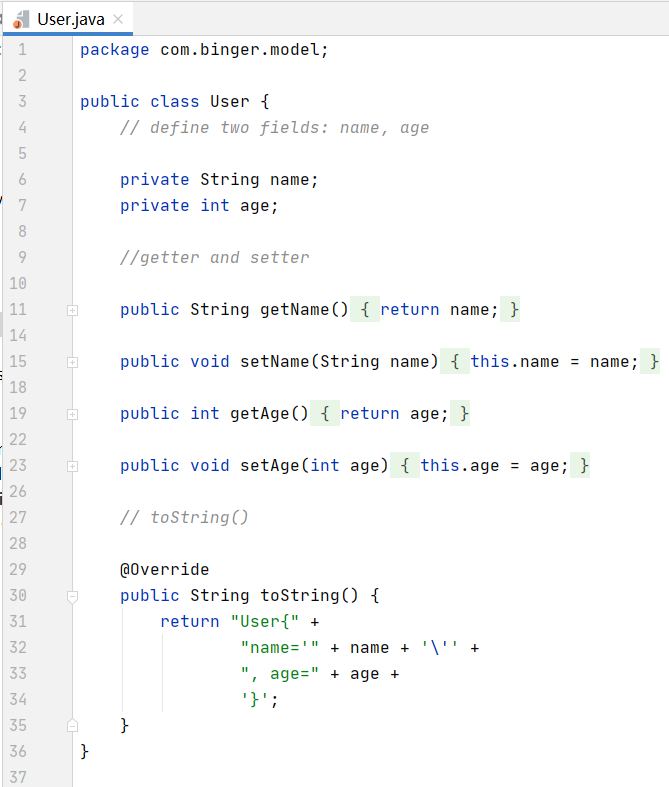


具体步骤：

1.在pom.xml中配置所需的jdk和依赖



2.在java目录下新建com.binger.model包，并创建一个User类，代码如下：



这是类中有两个私有属性，分别是name和age，还有一个方法toString；

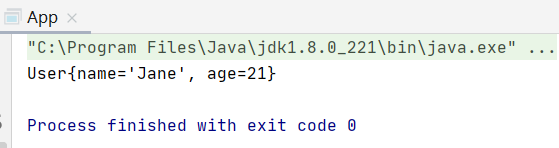
3.接下来配置applicationContext.xml文件



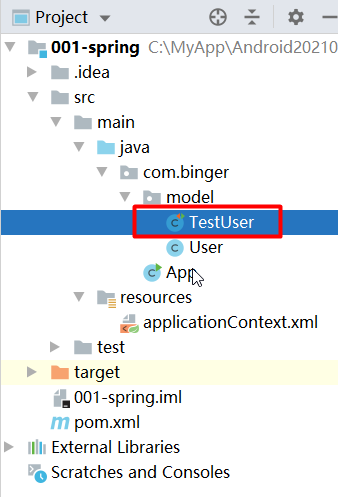
3.在App.java类进行测试



运行结果：

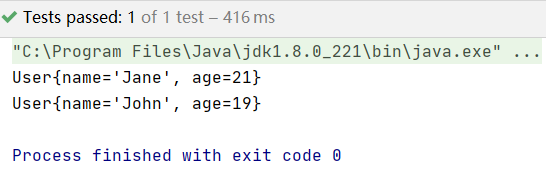


4.也可新建一个测试类进行测试





运行结果：



小结：这里我们并没有手动创建User的实例（对象），是Spring通过ApplicationContext帮我们创建的放在IoC容器里。ApplicationContext是一个IoC容器接口，它所创建的对象都称作是bean，也就是xml文件里的<bean id=" " class=" "></bean>这行配置信息。getBean方法就是从IoC容器里取得这个对象（根据标识id 和类名class），然后我们就可以调用该类的方法。

任务3：Spring注入

任务说明：用IDEA创建一个基于Spring Framework的程序，包名：com.binger ，项目名：002-spring

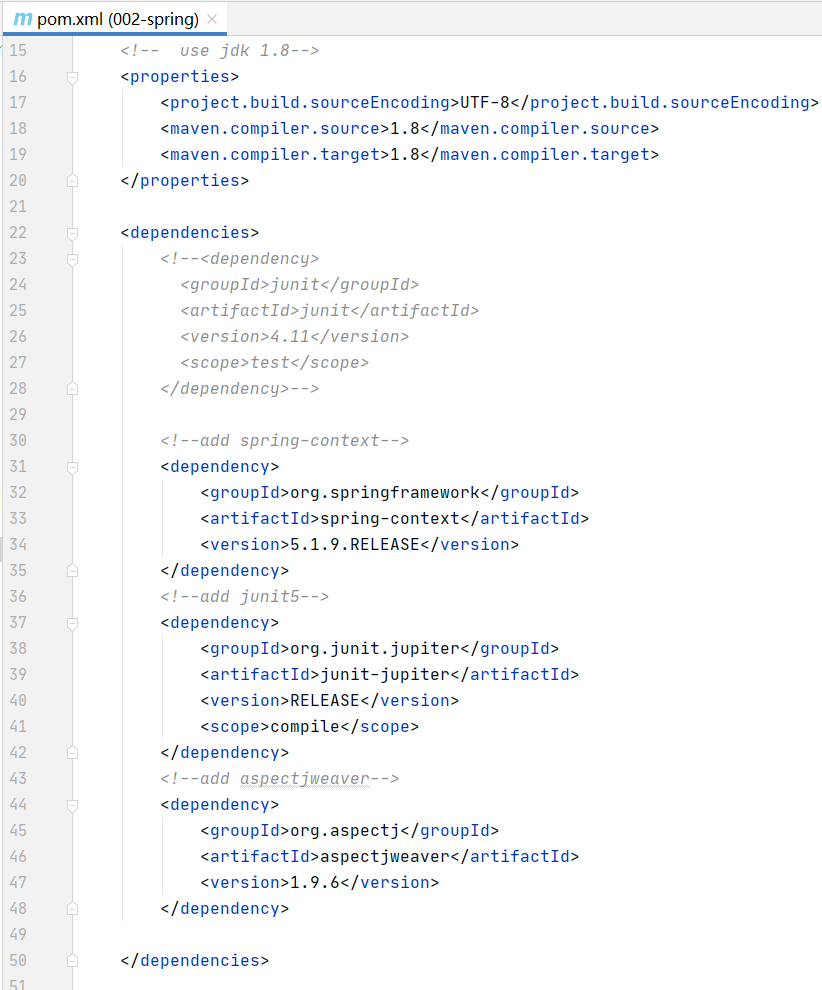
项目结构：



具体步骤：

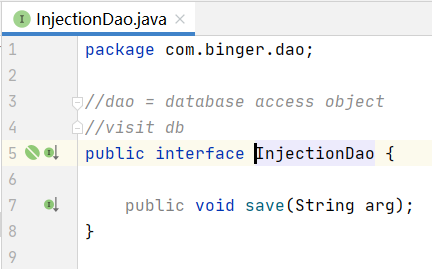
1. 在main文件夹中新建文件夹resources，将其设置为Resources Root，并在该文件夹中新建一个xml文件：spring-inject.xml

2.在pom.xml中配置jdk和依赖。

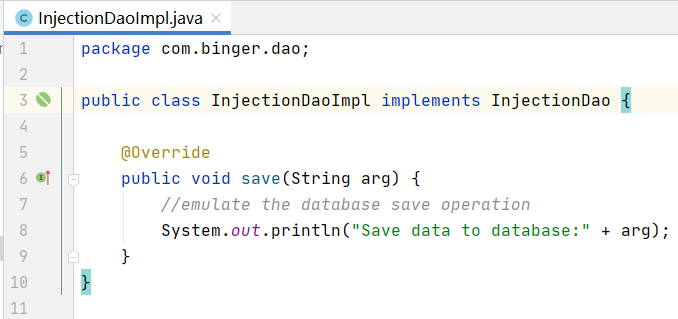


3.在dao包中的新建以下的接口和类，并编程。

InjectionDao代码：

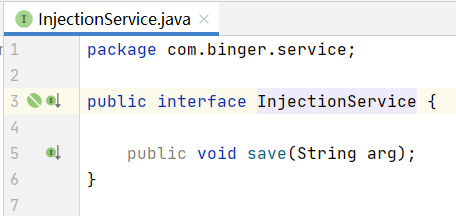


InjectionDaoImpl代码：



4.在service包中的新建以下的接口和类，并编程。

InjectionService代码：



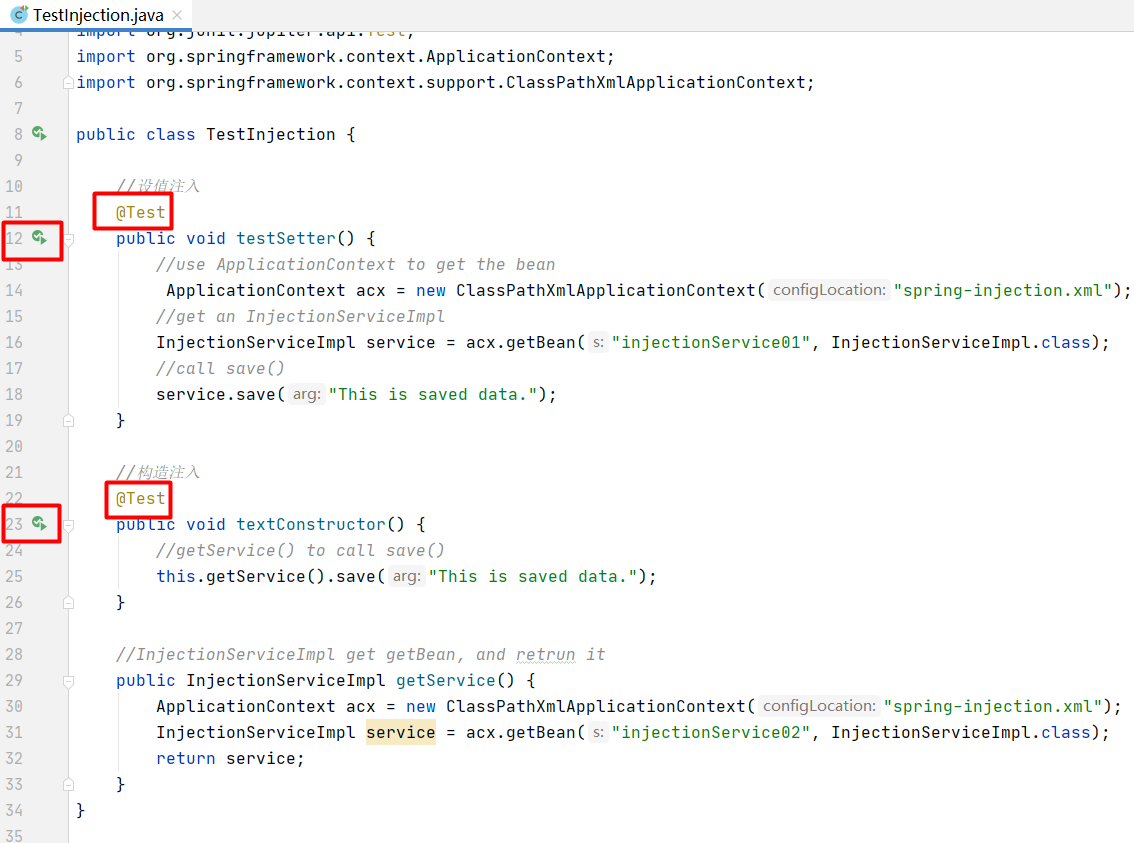
InjectionServiceImpl代码：



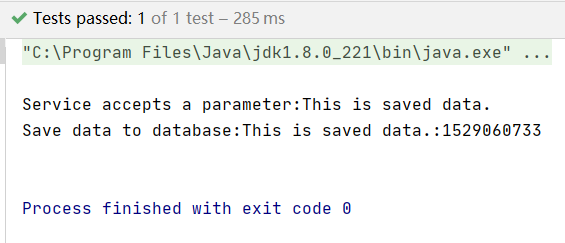
5.在spring-injection.xml文件中配置spring注入。



5.在test包中新建测试类：TestInjection，完成“设值注入”。



6.采用junit5测试，运行上图第12行的代码，运行效果：



7.“构造注入”与“设值注入”类似的操作，。





