**作业二——SpringBoot入门**

2019329621004 梅雨欣 19计算机科学与技术1班

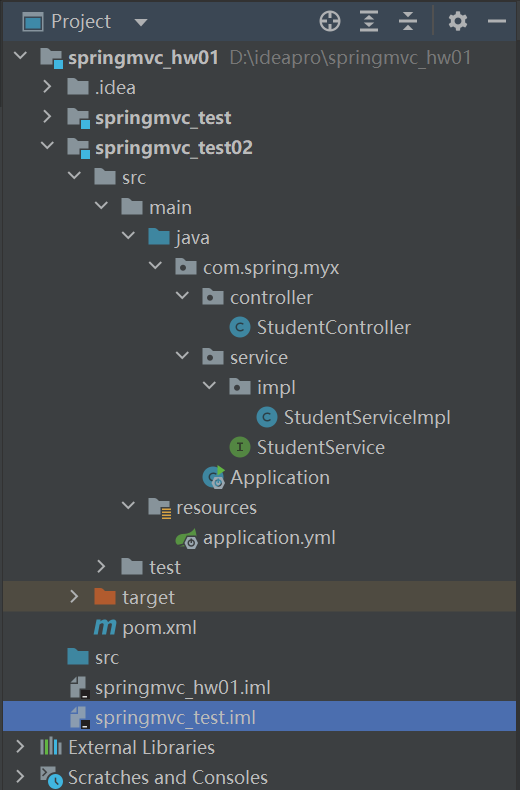
1. **题目描述**

参照Spring官网的Web app tutorial https://spring.io/guides/gs/rest-service/向导，实现当发送GET请求 http://localhost:8080/greeting请求，服务器接收请求并处理，响应JASON数据{"id":1,"content":"Hello, 姓名!"}；当发送带name参数的GET请求，如：http://localhost:8080/greeting/{学号}时，响应JASON数据{"id":2, "content":"Hello, 学号!"};

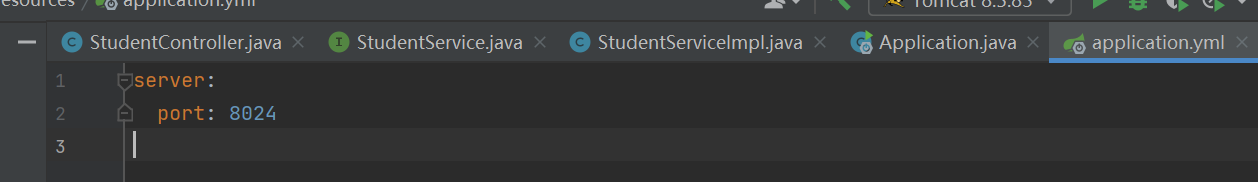
当发送POST请求http://localhost:8080/greeting时，其中request body的数据为JASON数据，格式为{"user":姓名,"pwd":学号}时，响应JASON数据{"id":3,"content":"Hello, 学号\_姓名!"};

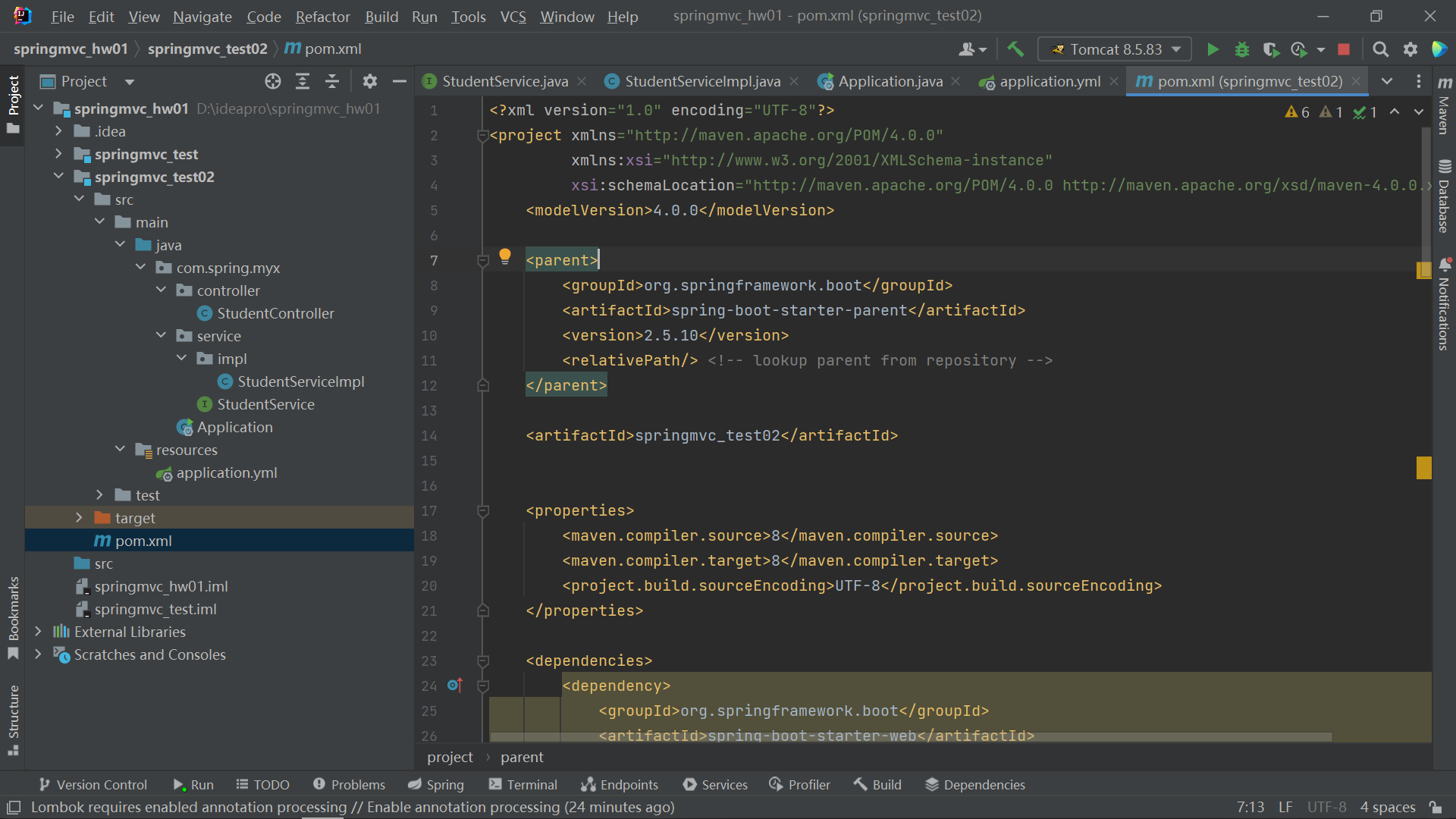
注意：题目中的学号和姓名实现时替换为自己的姓名和学号

1. **程序编写**
2. 还是在上次作业的项目springmvc\_hw01，再新建一个模块springmvc\_test02，在其中的java文件夹内创建包名com.spring.myx。项目目录如下：

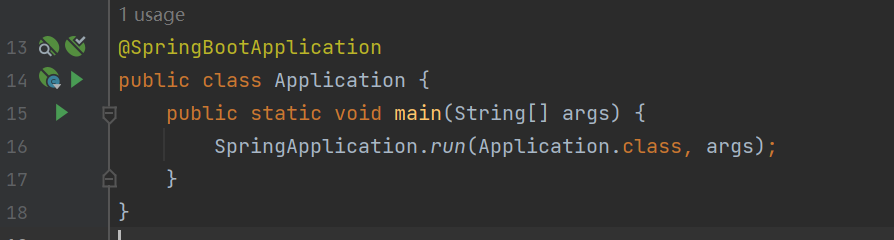


1. 配置文件（端口号），导入spring boot相关启动器。这里特别要注意不要用80、8080等常用端口号，会被占用。

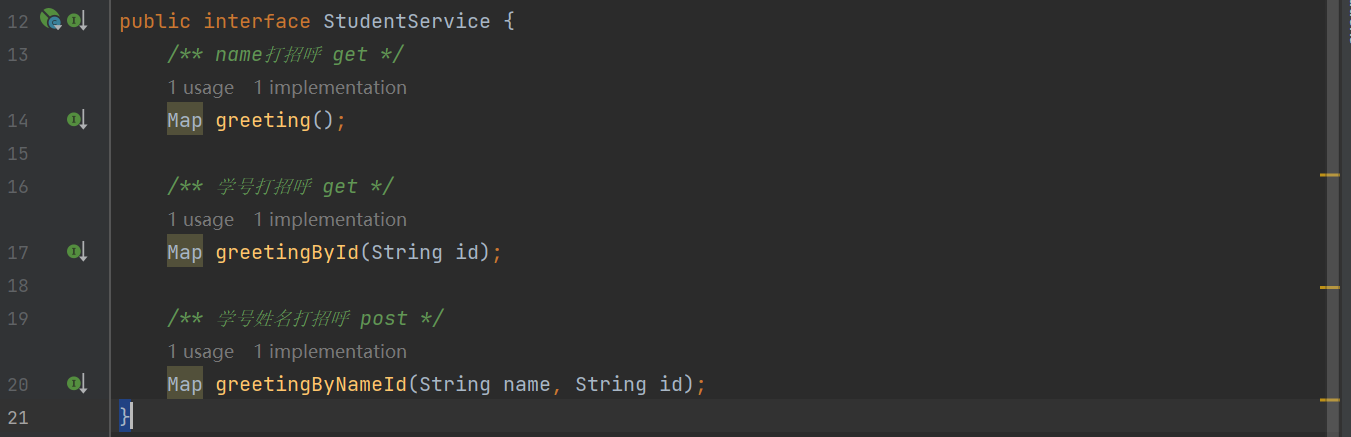


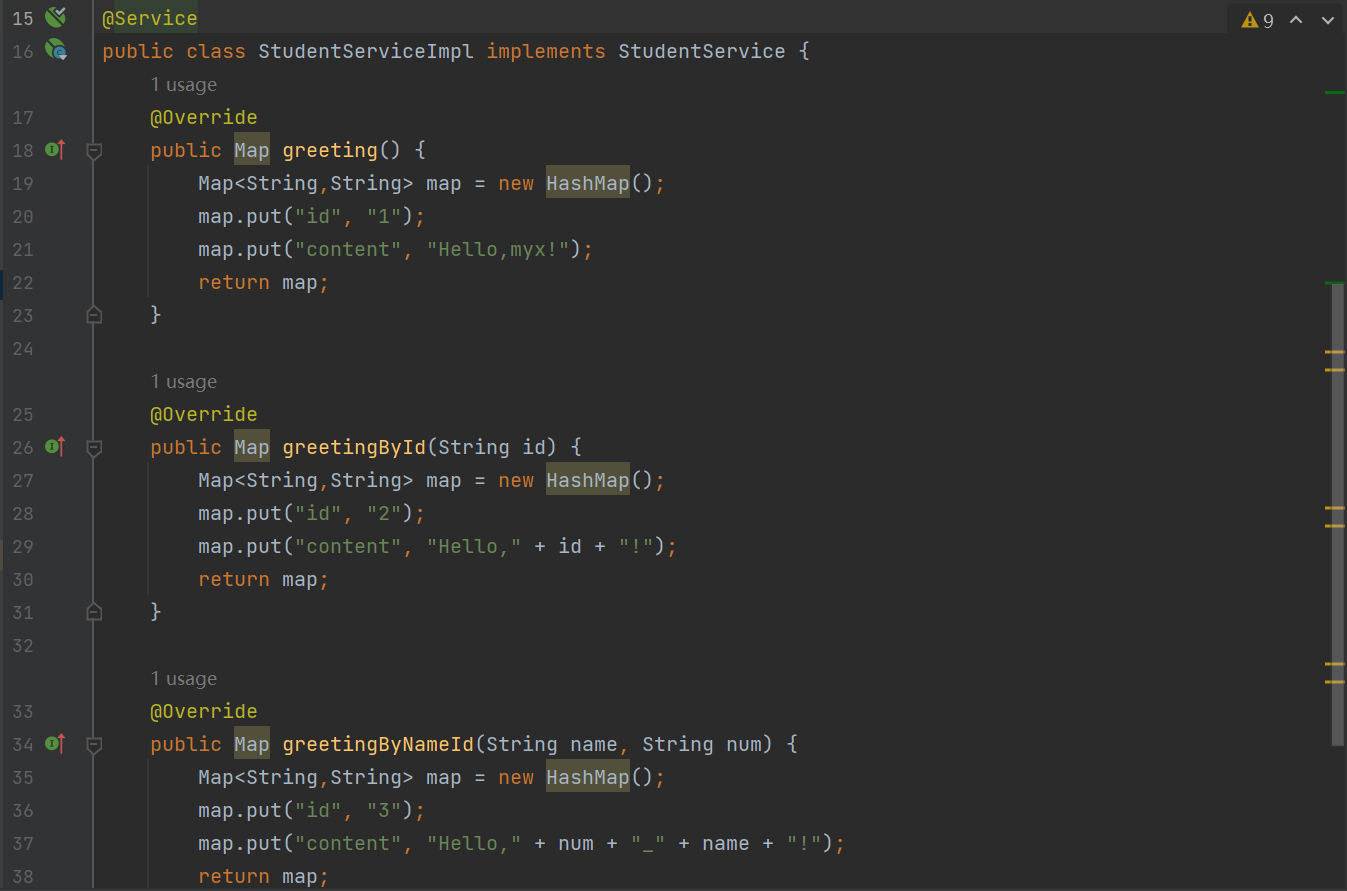


1. 编写SpringBoot启动类



1. 编写service层，实现接口。

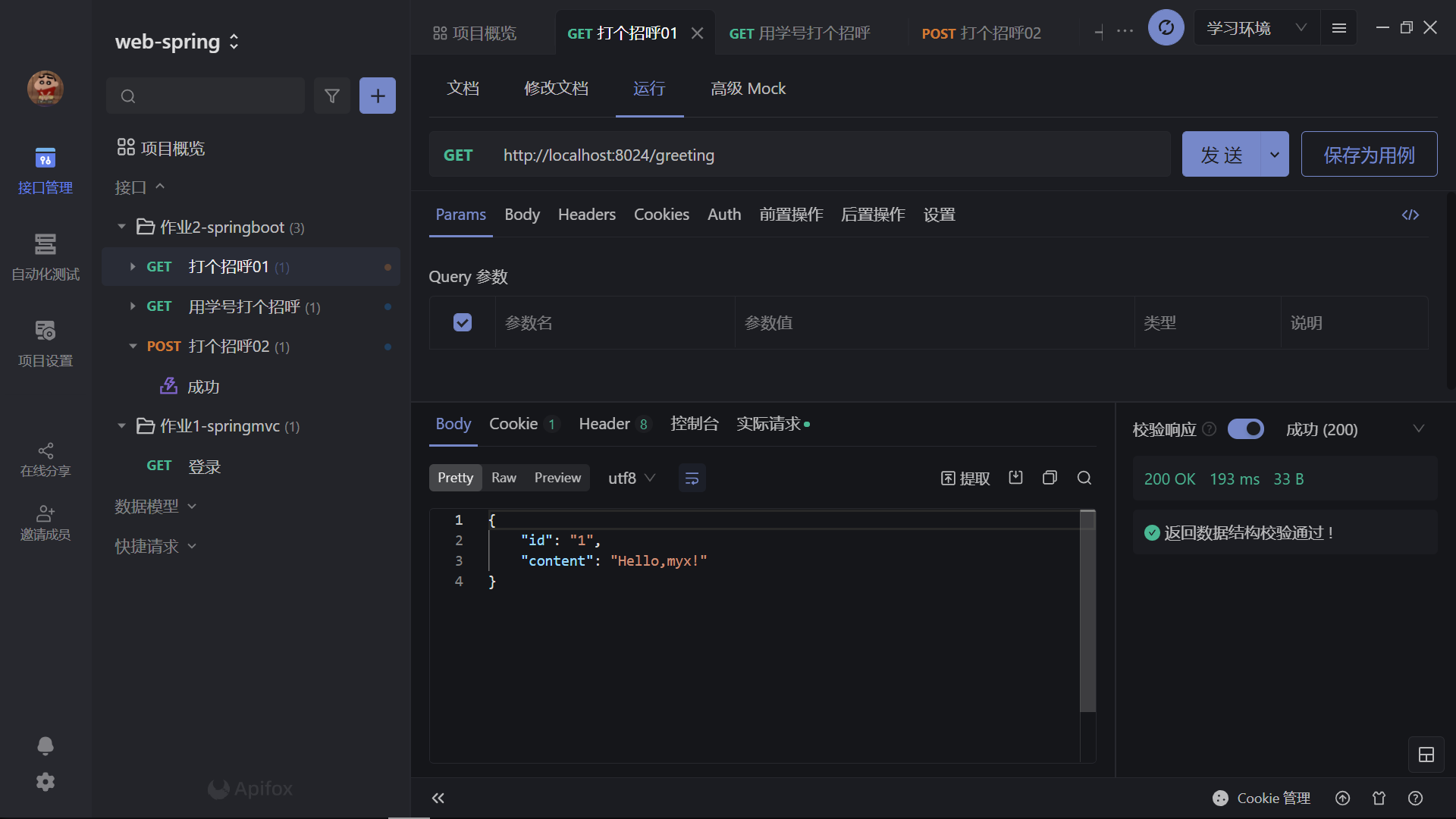




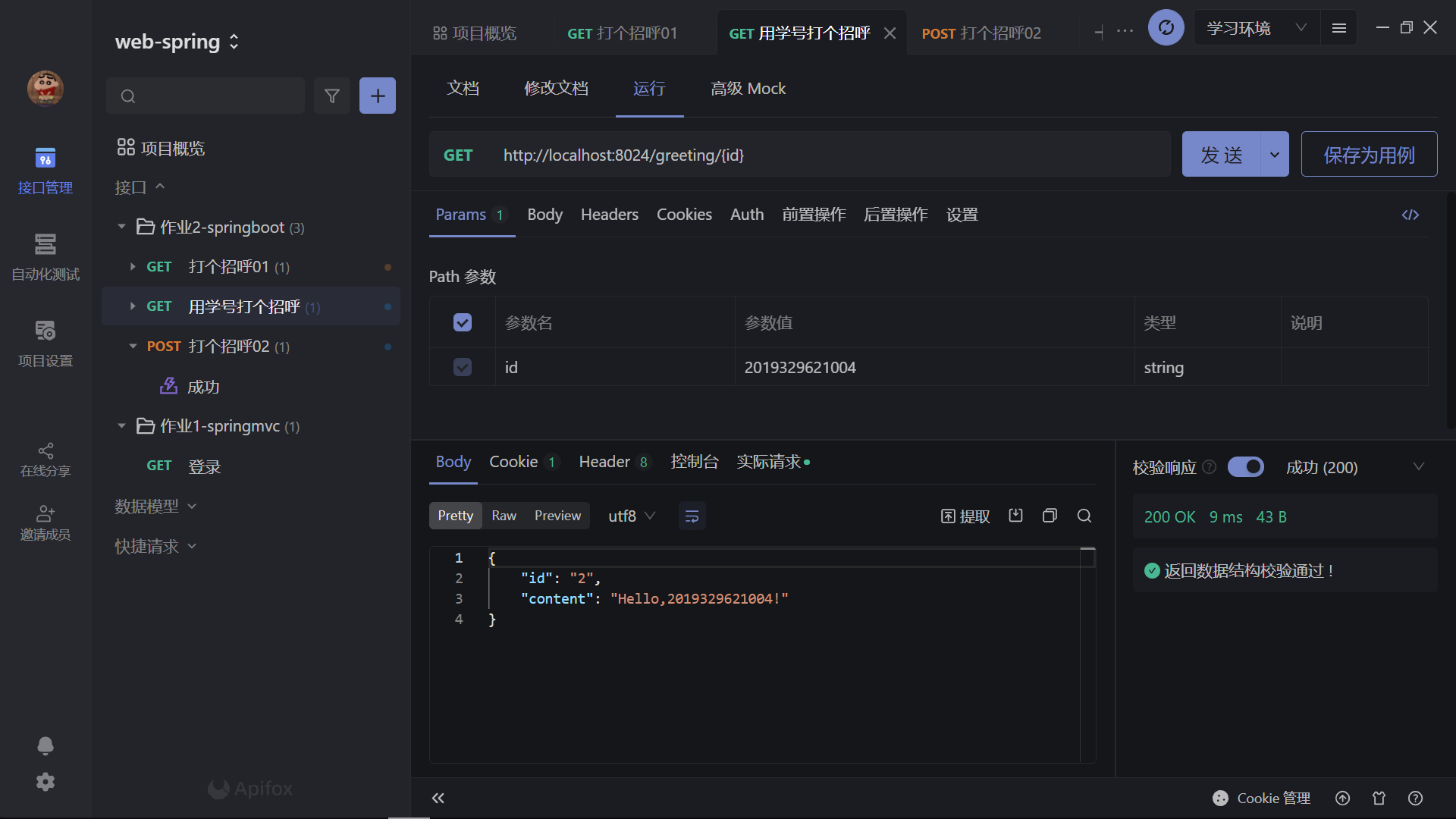
1. 编写controller层



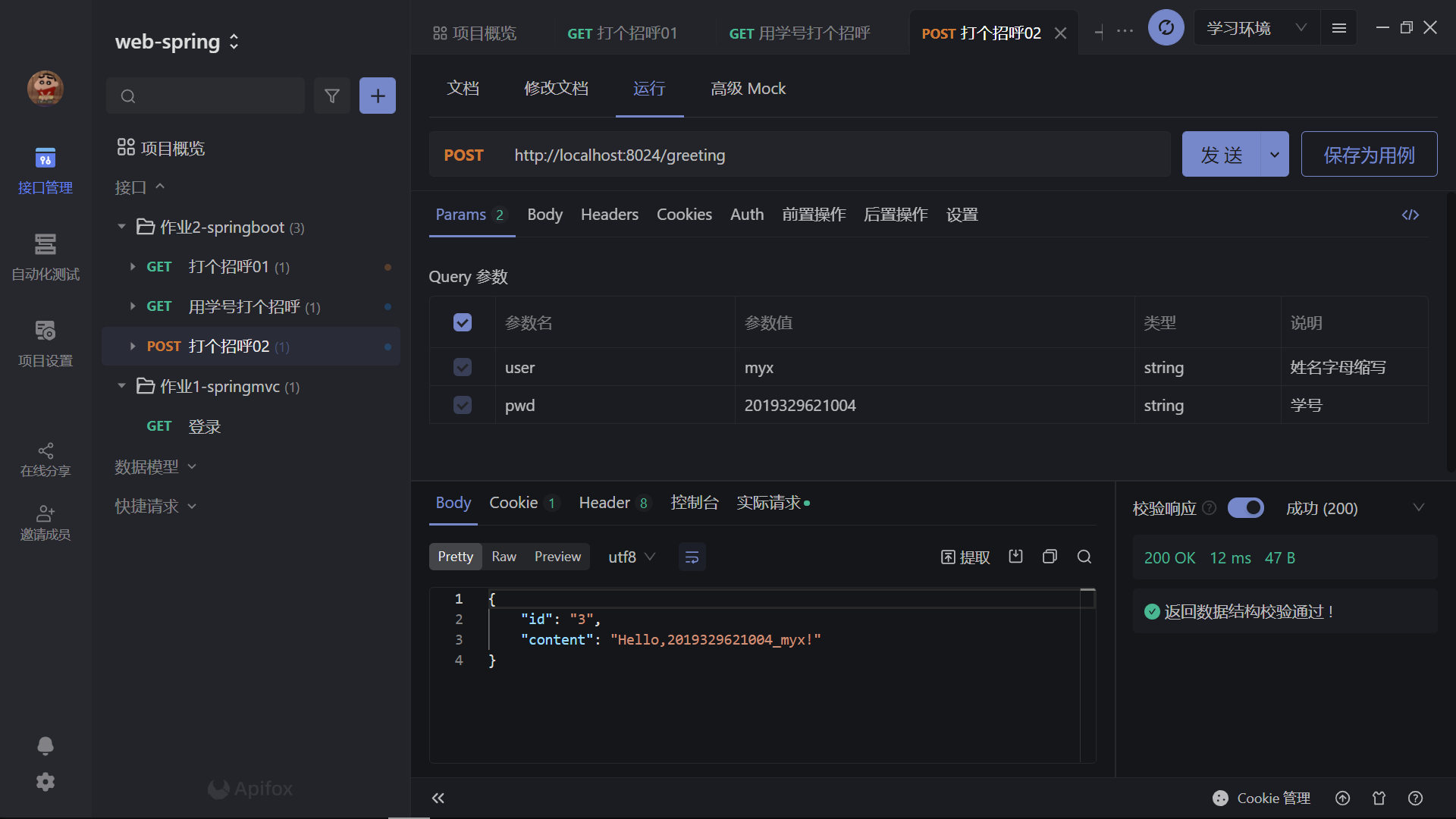
1. **执行结果**
2. 实现当发送GET请求 http://localhost:8024/greeting请求，服务器接收请求并处理，响应JASON数据{"id":1,"content":"Hello, 姓名!"}



1. 当发送带name参数的GET请求，如：http://localhost:8024/greeting/{id}时，响应JASON数据{"id":2,"content":"Hello, 学号!"};



1. 当发送POST请求http://localhost:8080/greeting时，其中request body的数据为JASON数据，格式为{"user":姓名,"pwd":学号}时，响应JASON数据{"id":3,"content":"Hello, 学号\_姓名!"};



1. **收获感受**

SpringBoot和上次的SpringMVC相比，搭建和配置都简单了非常多，不需要配置tomcat，只需要注意application.yml里的端口不能被占用，确实是官网上说的“just run”。这次还用到了ApiFox这个工具，非常好用，之前我是使用postman和swagger，后面发现它定位了postman+swagger+mock+jmeter，能力非常强大，使用也非常简单。