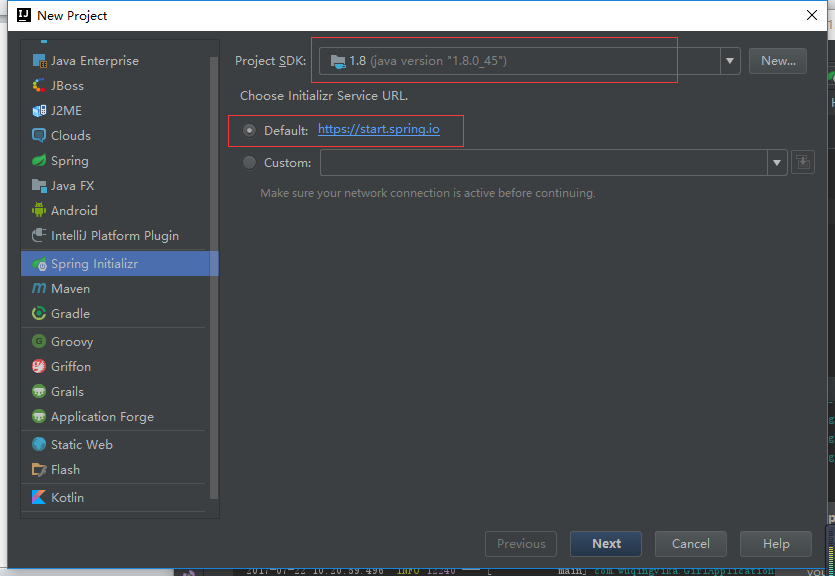
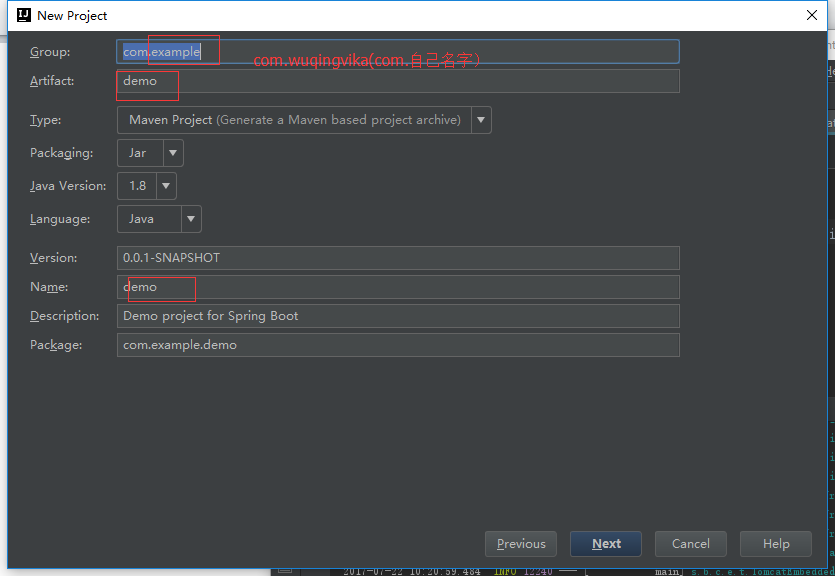
# 2017年7月22日星期六

## 1、初遇SpringBoot

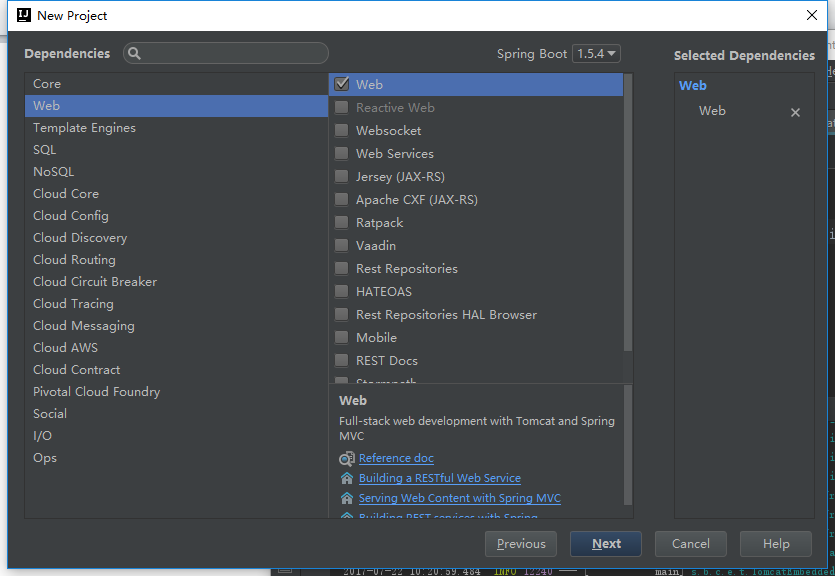
### 1.1 第一个SpringBoot应用



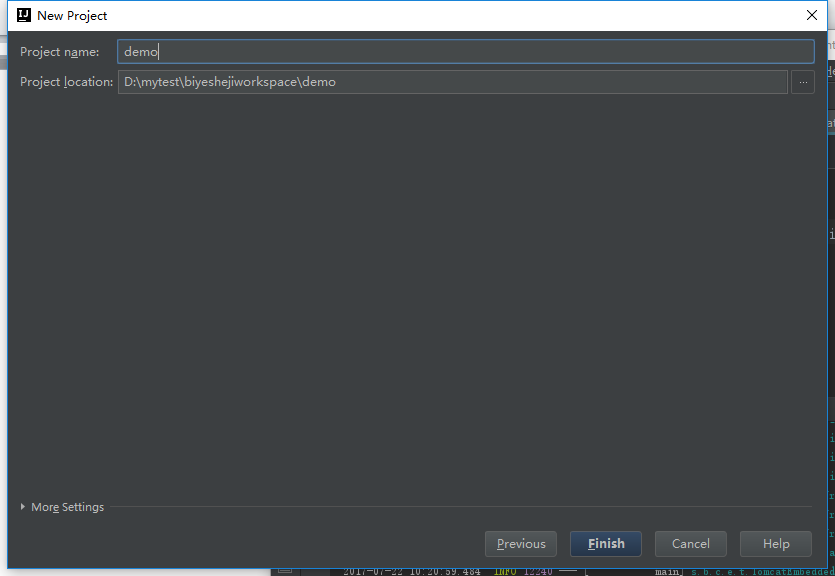
下一步：



然后 artifact随便 输入一个项目名，如girl..下面的name会自动变为girl.然后下一步吧。

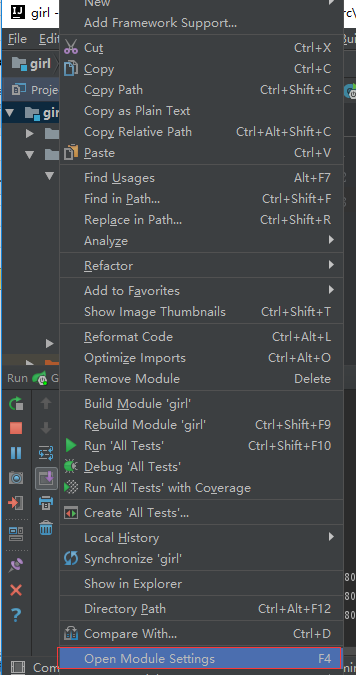


再下一步。

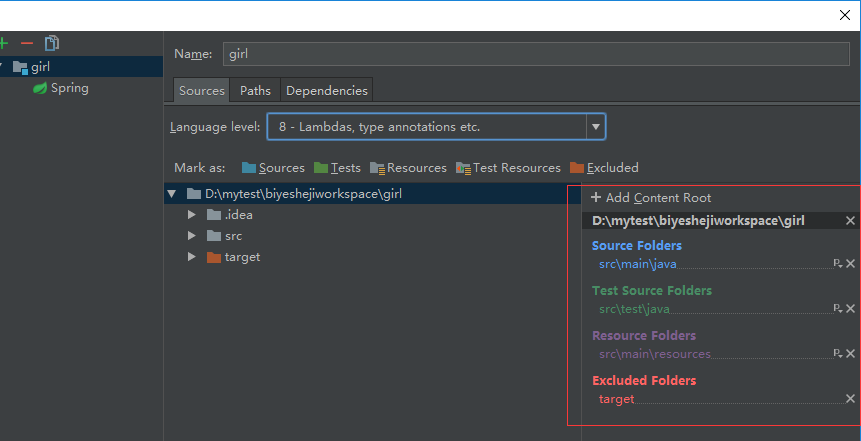


Finish。

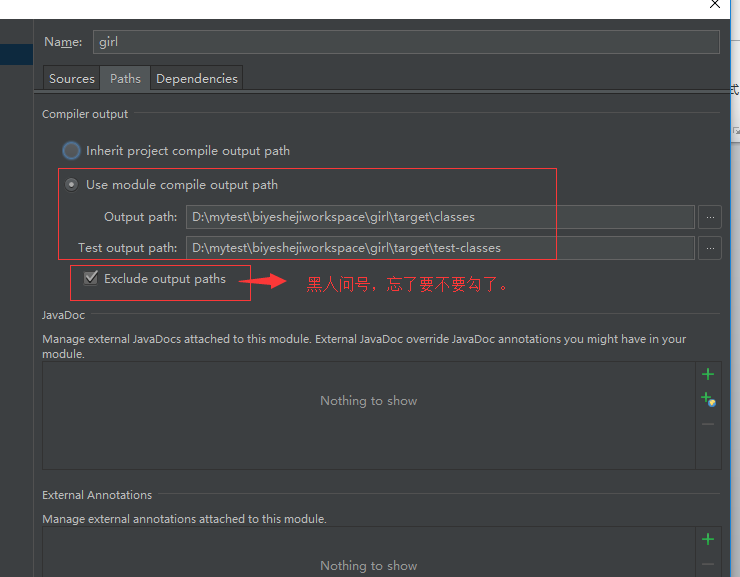
再右击你建的项目open module settings,如下：



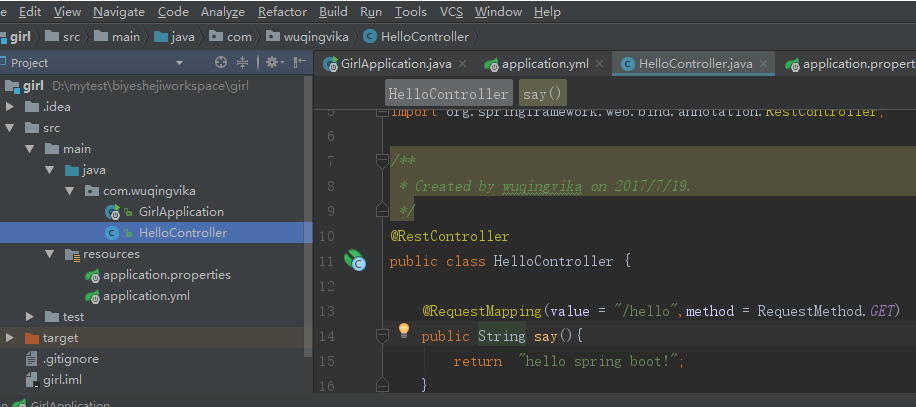
设置src(不设置不好右击run运行)



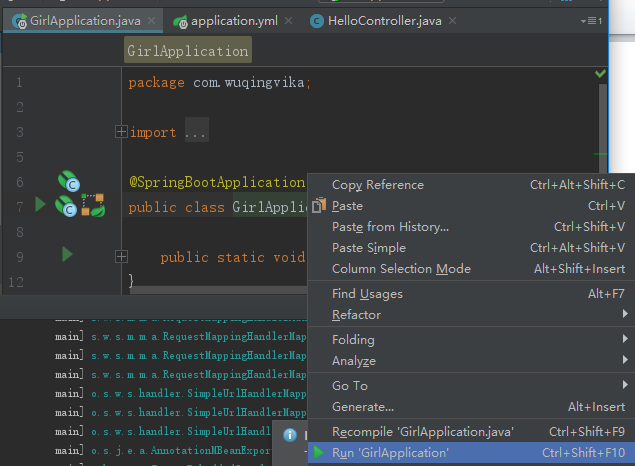
再点击Paths,设置



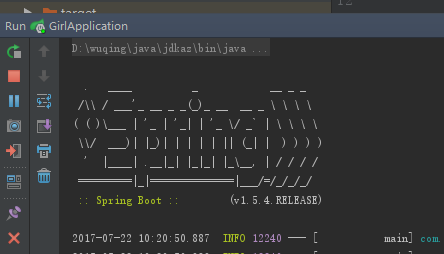
然后 添加 一个controller文件，



最后



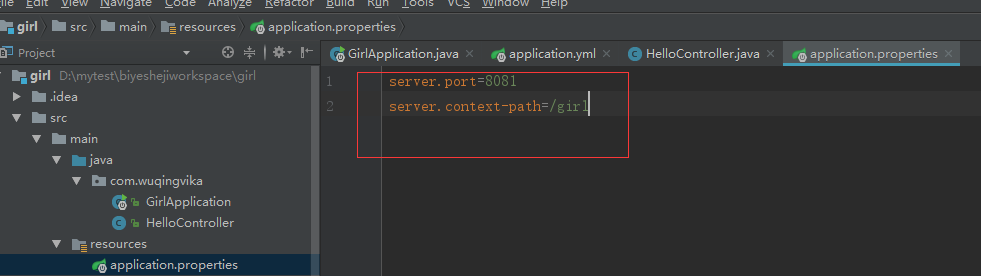
结果：



然后 浏览器访问http://localhost:8080/hello就行了

### 1.2 项目属性配置

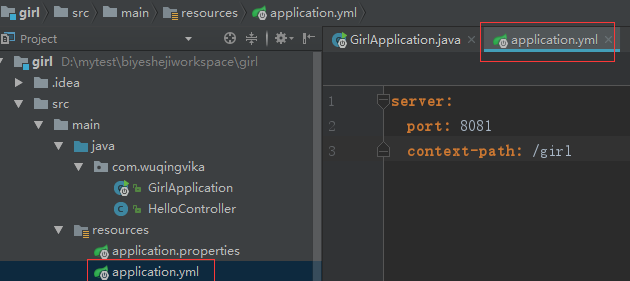
#### 1.2.1 配置端口



然后重新启动， 浏览器访问http://localhost:8081/girl/hello就行了

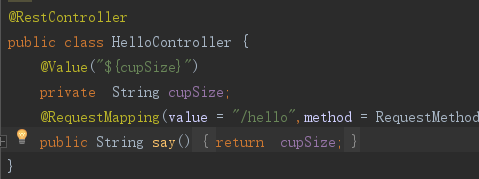
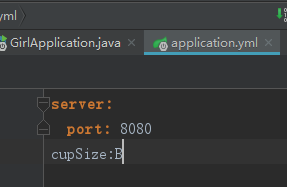
#### 1.2.2 简化版配置文件(比较常用也推荐使用)

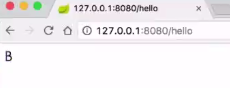
创建application.yml,配置如下 相同前缀可用Tab错开，注意冒号之后留有一个空格。然后删掉application.properties文件。



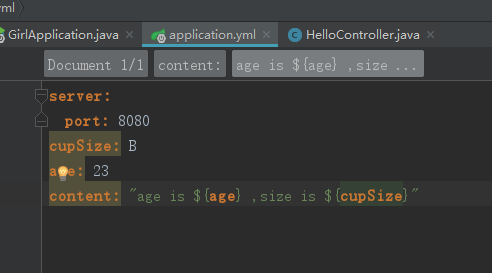
#### 1.2.3 配置自定义属性

（1）cupSize为自定义属性。



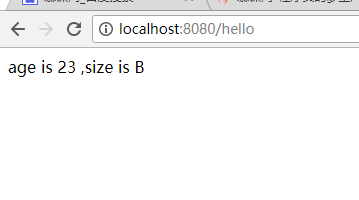


#### 1.2.4 自定义多个属性

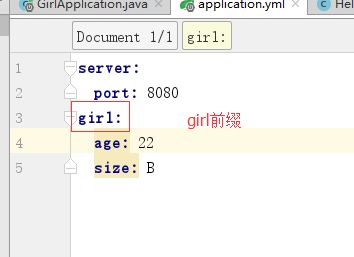


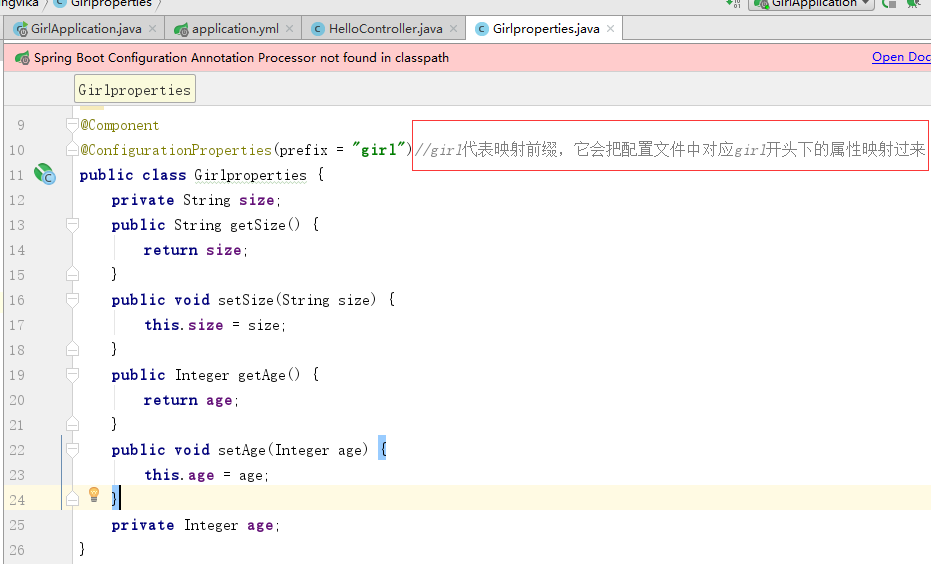


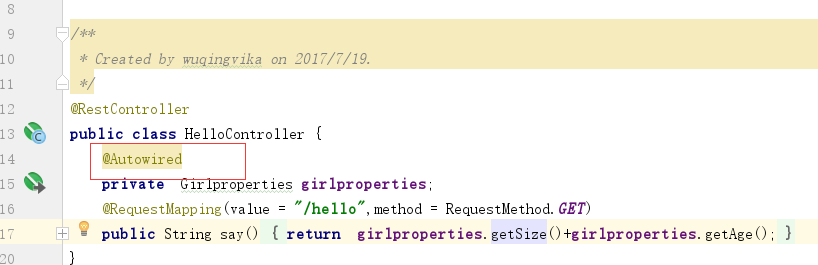
访问结果：



#### 1.2.5 封装类

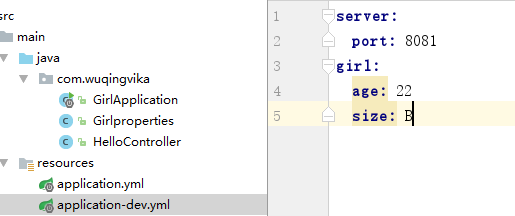


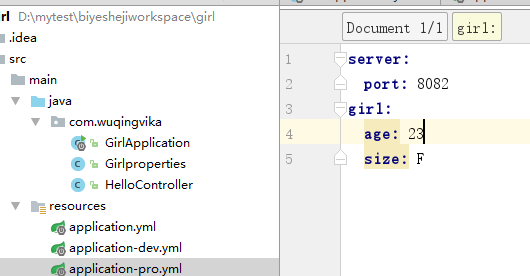




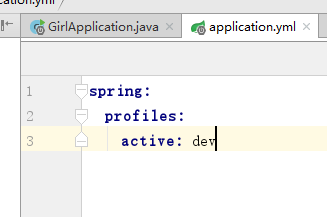
#### 1.2.6 切换配置文件

1.如果说开发环境是尺寸为B，端口为8081，生产环境为F端口为8082。

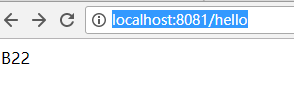




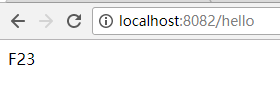
那么怎么在两个环境中切换呢，只需要将application.yml定义激活为对应的值就好。



如上已经配置为开发环境，则在浏览器中访问对应端口结果如下



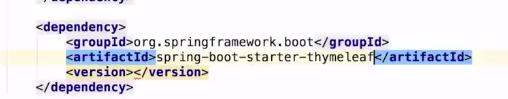
相反如果为生产环境，则结果为：



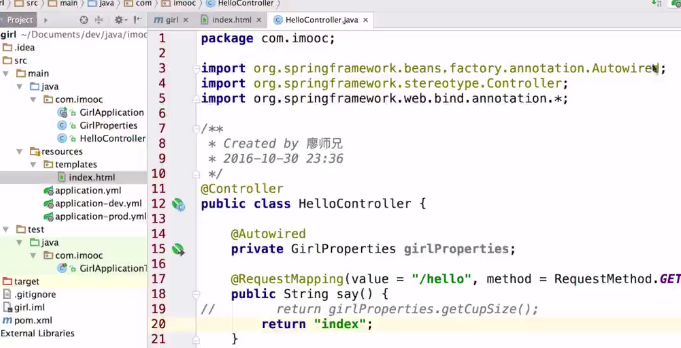
### 1.3 Controller的使用

#### 1.3.1 Controller

1 添加组件



在resource下添加一个templates包。在这个目录下，新建一个index.html.



RestController等同于controller+ResponseBody



#### 1.3.2 URL映射

请求可以为hello或hi



也可以在类上面写，访问时只需要拼接就行(/hello/say)



#### 1.3.3 设置请求参数



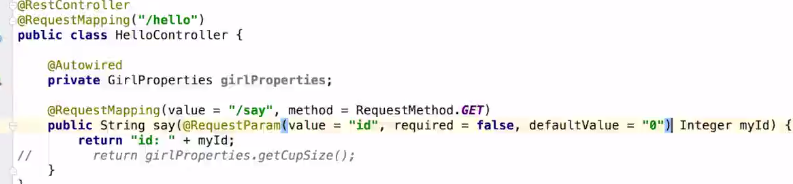
(1)pathvariable



(2)RequestParam 这时只要引号内Id取名一样就行



扩展（是否为必传：false，默认值为0）

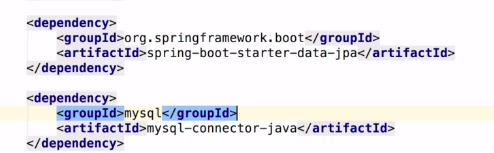


简化RequestMapping,有GetMapping(推荐),还有PostMapping等组合注解。

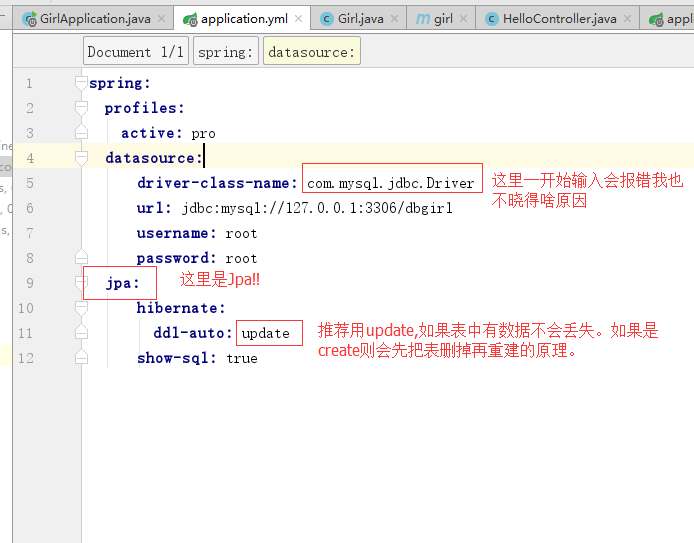


### 1.4 数据库操作

#### 1.4.1 添加组件



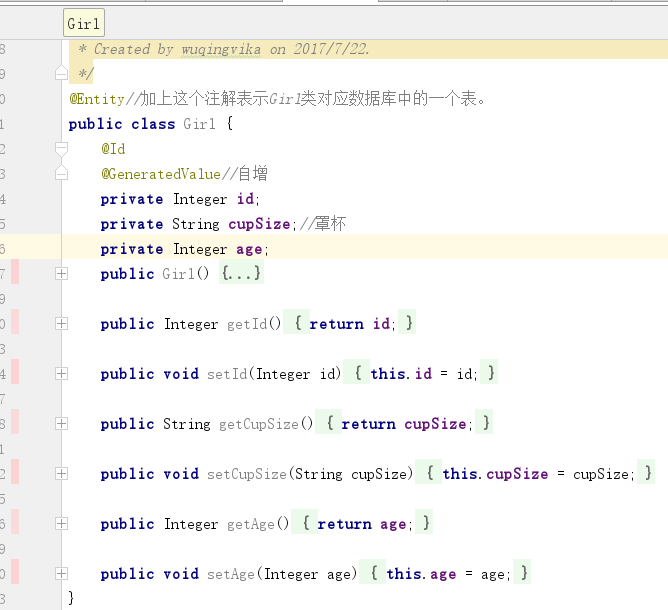
#### 1.4.2 配置文件



注意，数据库先得建立好。

#### 1.4.3 配置实体类

记得添加构造方法，以及get/setter不然报错

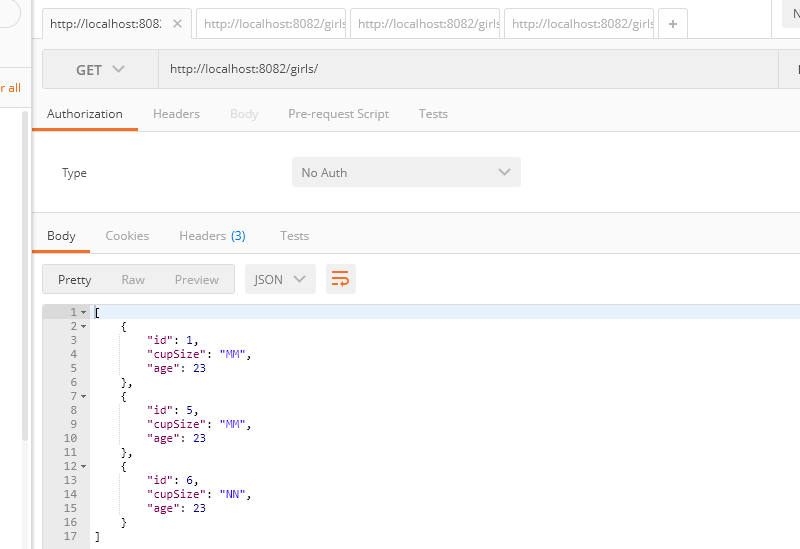




#### 1.4.4 获取女生列表

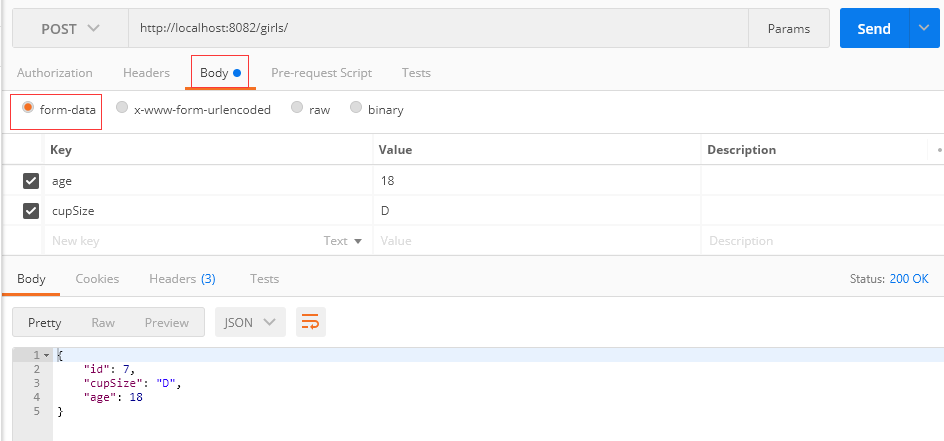


结果



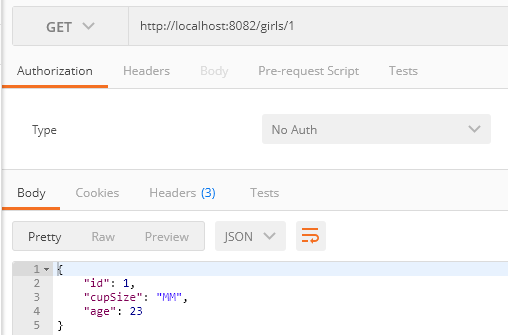
#### 1.4.5 创建一个女生

@PostMapping(value = **"/girls"**)  
**public** Girl girlAdd(@RequestParam(**"cupSize"**) String cupSize,  
 @RequestParam(**"age"**) Integer age){  
 Girl girl=**new** Girl();  
 girl.setAge(age);  
 girl.setCupSize(cupSize);  
 **return girlRepository**.save(girl);  
}



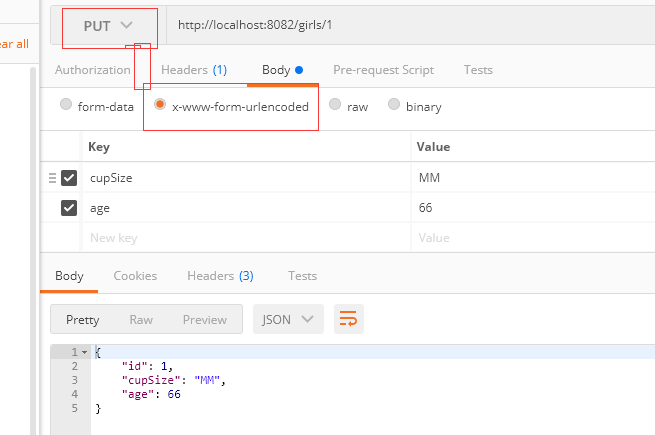
#### 1.4.6 通过id查询一个女生

@GetMapping(value = **"/girls/{id}"**)  
**public** Girl getGirlById(@PathVariable(**"id"**) Integer id){  
 **return girlRepository**.findOne(id);  
}



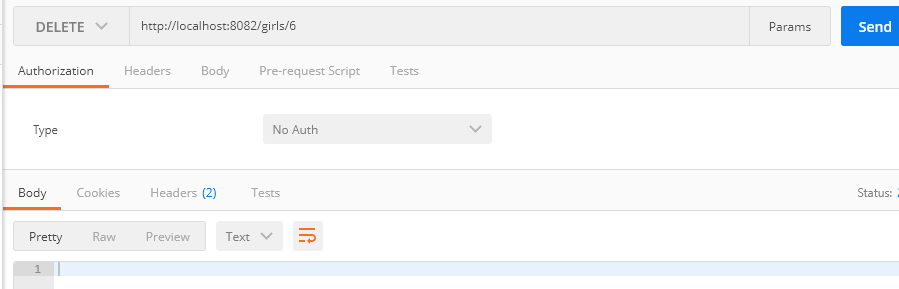
#### 1.4.7 通过id更新一个女生

@PutMapping(value = **"/girls/{id}"**)  
**public** Girl girlUpdate(@PathVariable(**"id"**) Integer id,@RequestParam(**"cupSize"**) String cupSize,  
 @RequestParam(**"age"**) Integer age){  
 Girl girl=**new** Girl();  
 girl.setId(id);  
 girl.setAge(age);  
 girl.setCupSize(cupSize);  
 **return girlRepository**.save(girl);  
}

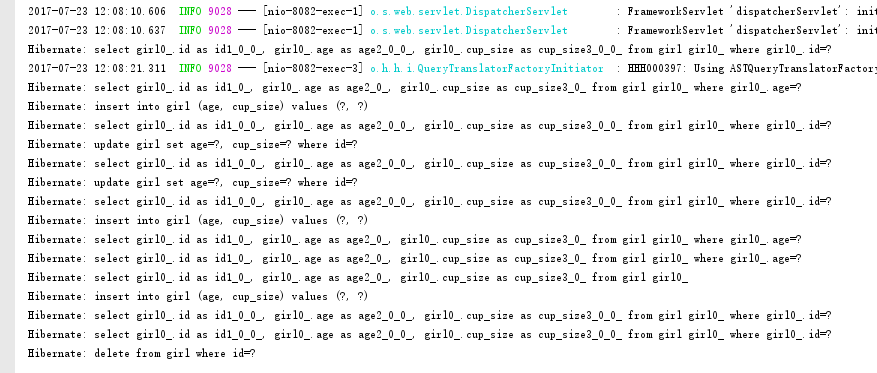


#### 1.4.8 通过id删除一个女生

@DeleteMapping(value = **"/girls/{id}"**)  
**public void** girlDel(@PathVariable(**"id"**) Integer id){  
 **girlRepository**.delete(id);  
}



控制台输入的sql



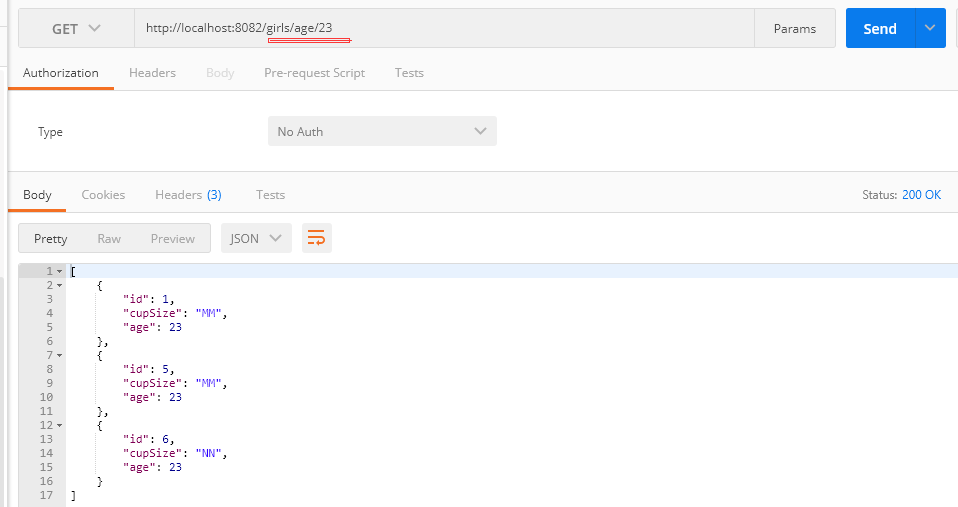
#### 1.4.9 通过age获取女生列表

@GetMapping(value = **"/girls/age/{age}"**)  
**public** List<Girl> findByAge(@PathVariable(**"age"**) Integer age){  
 **return girlRepository**.findByAge(age);  
}

注意扩展方法的命名规则，详情见图：

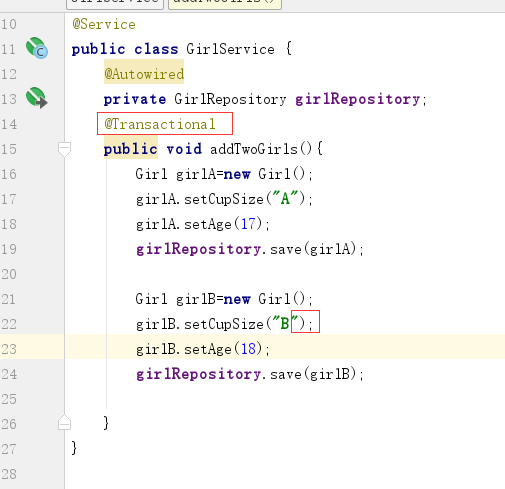


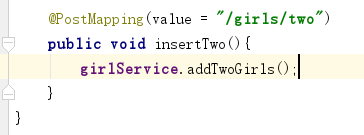
结果如下：



### 1.5 事务

添加两个女生，要么一起添加成功，要么一起添加失败。





然后慢慢测试吧。

## 2、PostMan安装使用

1.下载postman，Postman-REST-Client.zip（<http://download.csdn.net/download/u010246789/9528471#comment>）

2.解压后找到文件Postman-REST-Client\_v0.8.4.14.crx

3.将Postman-REST-Client\_v0.8.4.14.crx后缀名改为rar

4.将Postman-REST-Client\_v0.8.4.14.rar解压缩

5.将解压缩后文件夹中\_metadata改为metadata

6.打开chrome更多工具-->扩展程序，点击开发者模式，显示加载已解压的扩展程序

7.点击加载已解压的扩展程序，选中第4步解压缩的文件夹即可

8.点击安装完成后的详细信息，可以创建快捷方式，而后双击即可使用。

具体见 <https://wenku.baidu.com/view/7339bcd9580216fc710afd99.html>

心累搞了好久呢

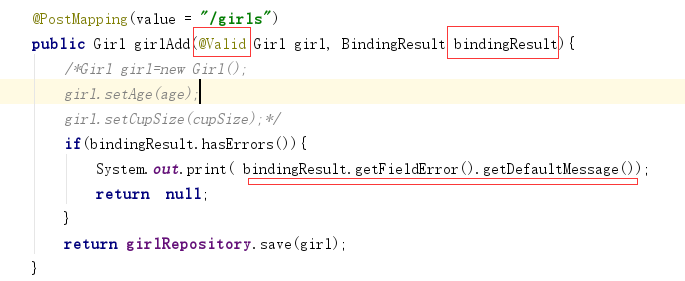
后来直接在google应用里添加应用就ok的。

## 3、SpringBoot Web篇

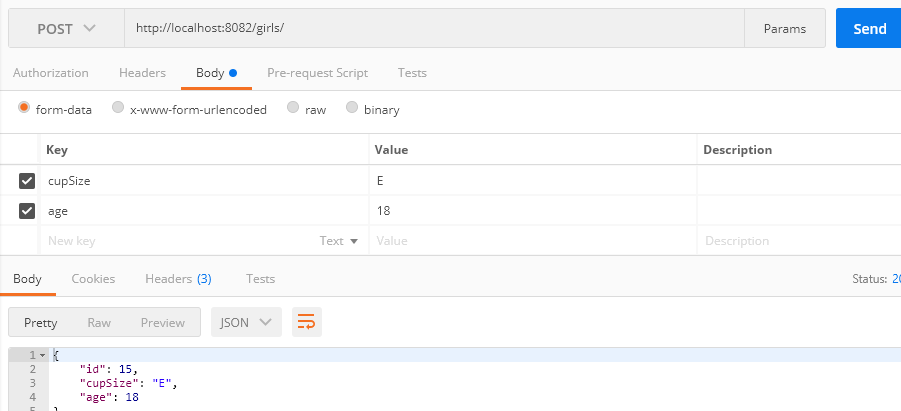
### 3.1 表单验证

验证年龄小于18不能添加

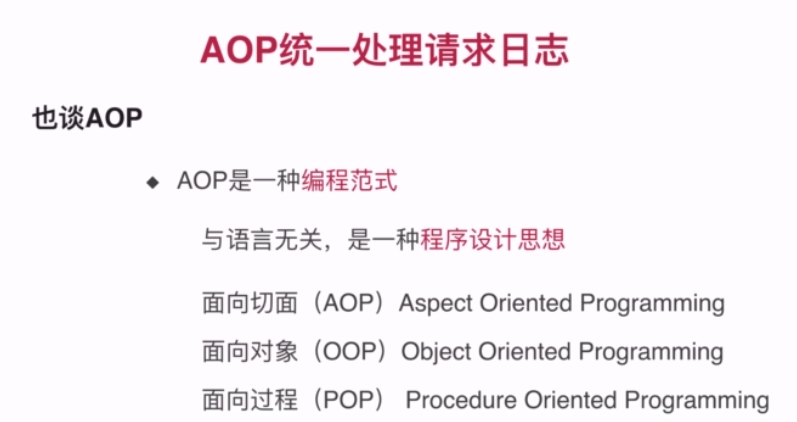




结果（成功）

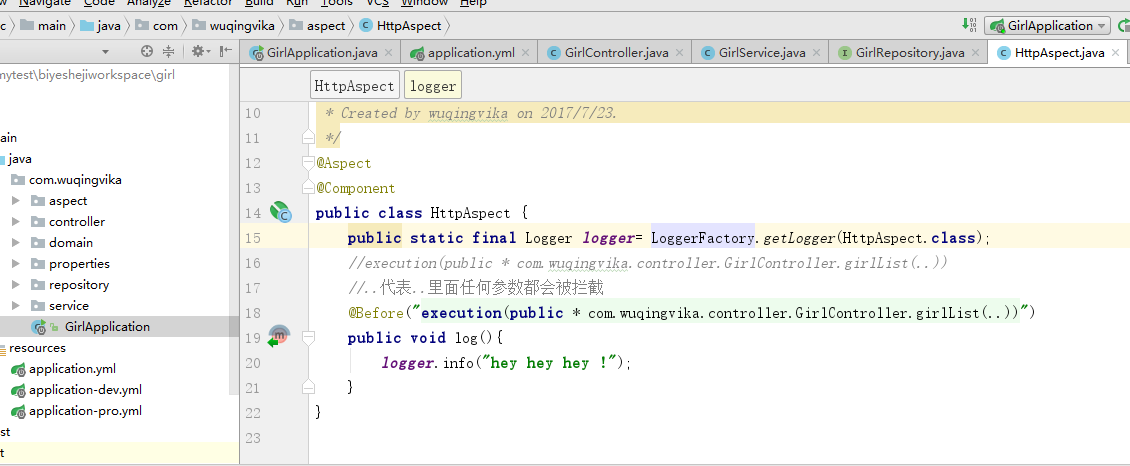


### 3.2 使用Aop处理请求

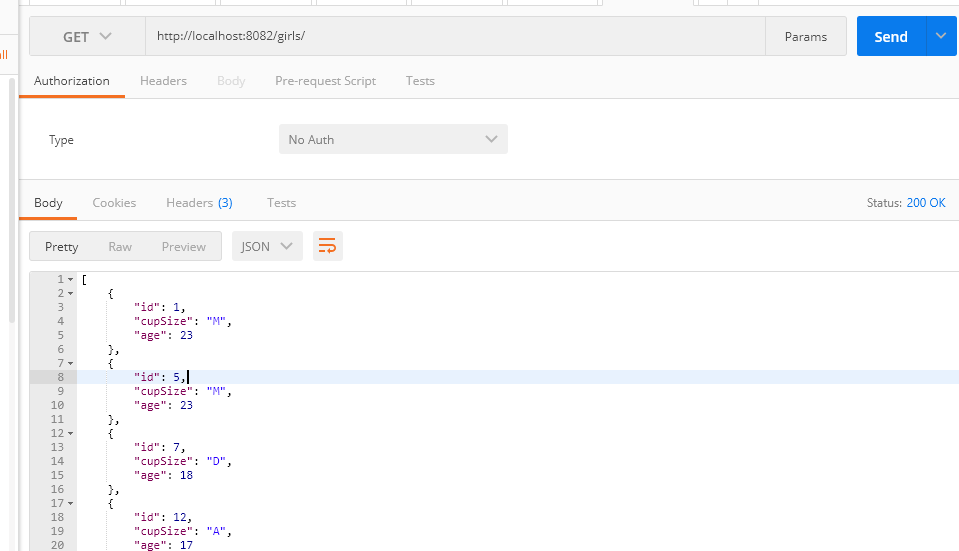


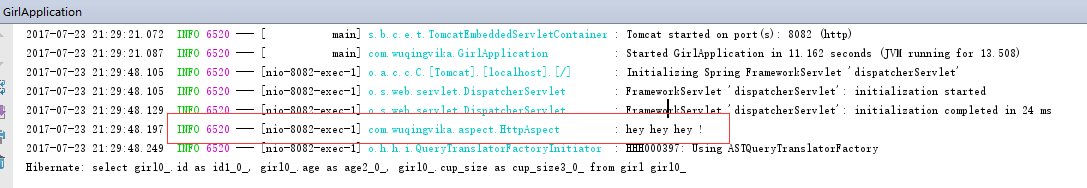


#### 3.2.1 拦截一个方法









#### 3.2.2 拦截所有方法

拦截GirlController中所有方法。@Before(**"execution(public \* com.wuqingvika.controller.GirlController.\*(..))"**)

