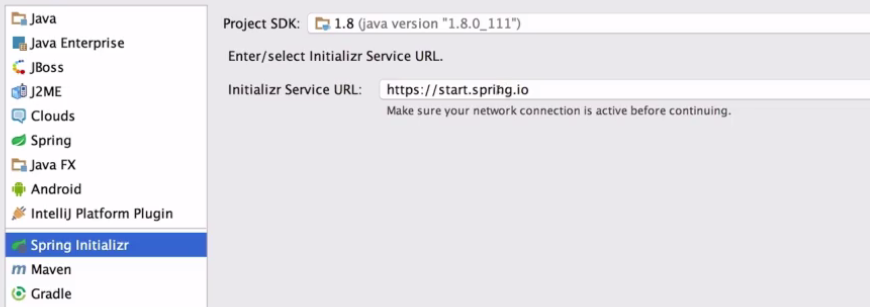
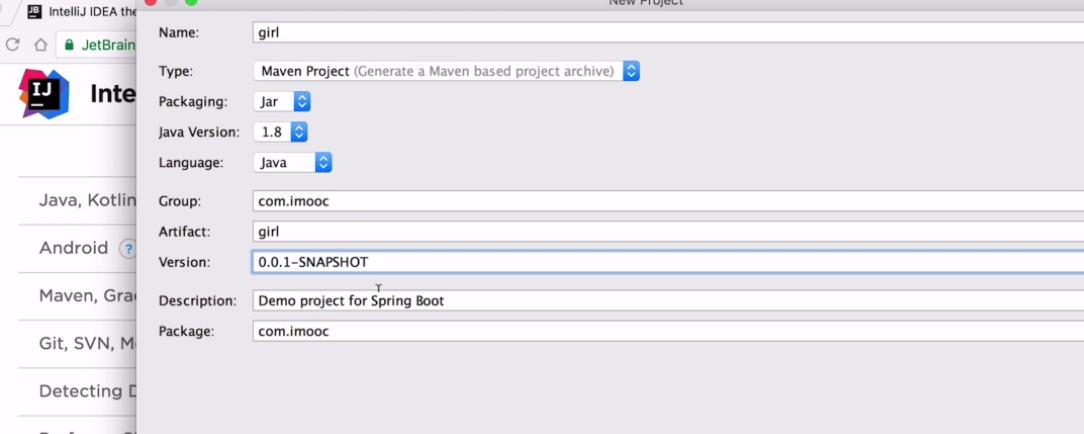
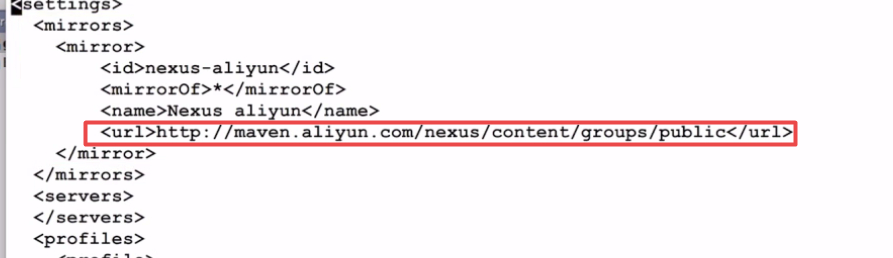
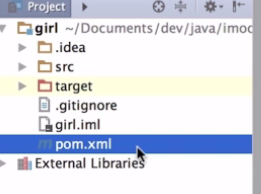
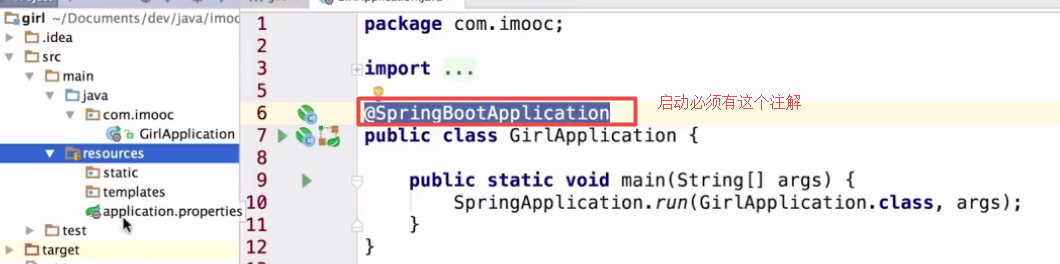
Postman工具（restful测试工具）











启动方式：

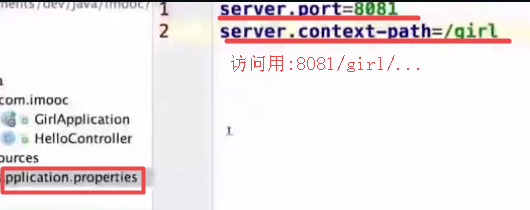
第一种：注解在@SpringBootApplication注解下的main方法启动

第二种：

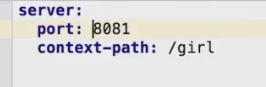
第三种：maven install编译

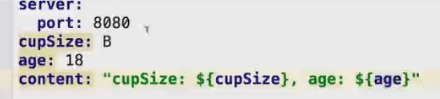
直接使用java命令启动

配置文件，这里配置后就自动读取，不需要其他操作

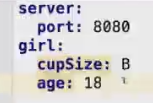


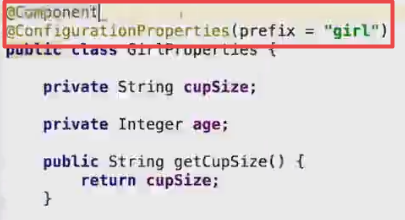
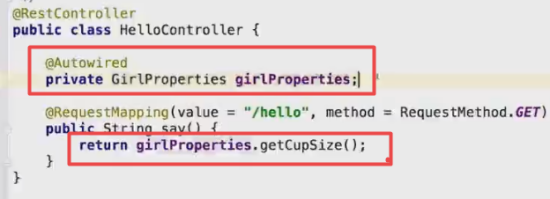
Spring boot最好使用yml配置文件，冒号后面要加空格(语法)



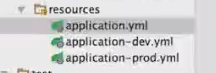
在配置里面使用当前配置

@value()注入yml中的属性

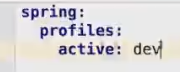
获取yml中girl对象

获取前缀是girl的配置，这里必须使用@Component才能添加到bean  


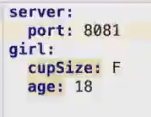
灵活配置开发/生产环境

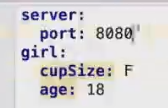


application.yml配置为开发环境，active为哪个就调用哪个配置，灵活控制



application-dev.yml配置开发环境

  
application-prod.yml配置生产环境



启动时灵活配置启动的yml，active是哪个就以哪个yml启动



Maven配置



@RestController = @Controller+@ResponseBody

请求最好加上请求方法类型：method=RequestMethod.GET

@RequestMapping(value={“/hello”,”/h1”}, method=RequestMethod.GET)

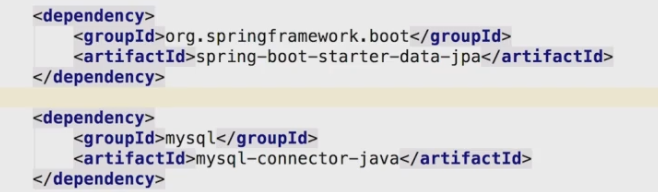
@PathVariable 获取url中的数据

@RequestParam 获取请求参数的数据(get/post均可获得)

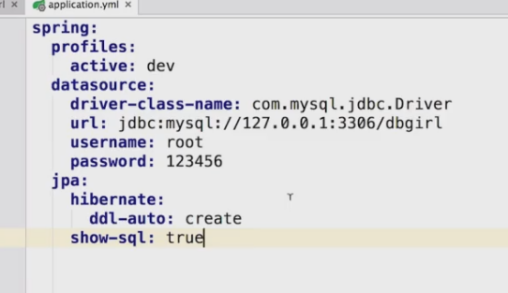
@GetMapping @PostMapping组合注解

Spring-Data-Jpa对Hibernate的整合

使用数据库需要添加两个组件



application.yml



Ddl-auto:create 每次运行重新创建(删除之前的表)

Ddl-auto:update 每次运行不删除表，没有就创建

@Entity表示对应数据库里面的一个表

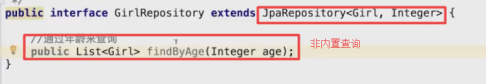
@Id表示自增

@GeneratedValue 表示主键

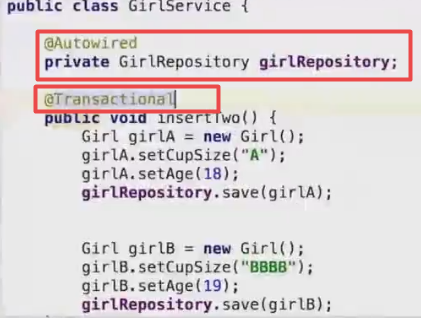


操作数据库



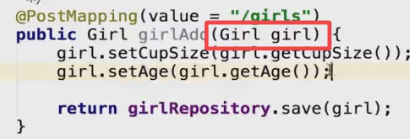


事务管理

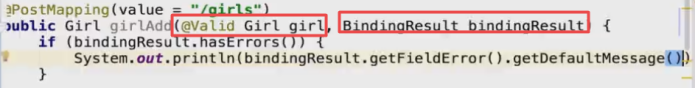


使用@Transactional控制事务，一条插入也可能失败，最好都加上事务控制，只有查询可以不用事务

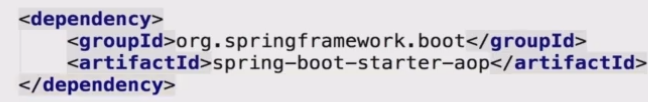
**表单验证**

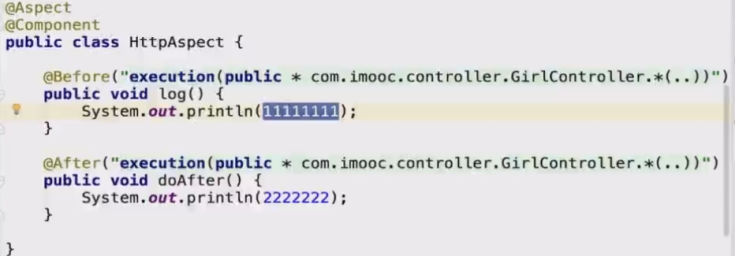


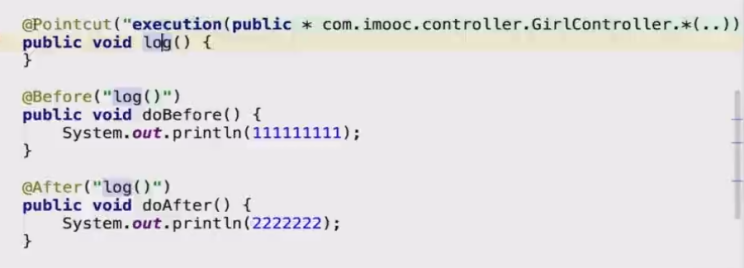


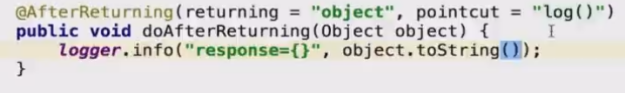


AOP同意处理请求日志(面向切面)









@Aspect 切面注解

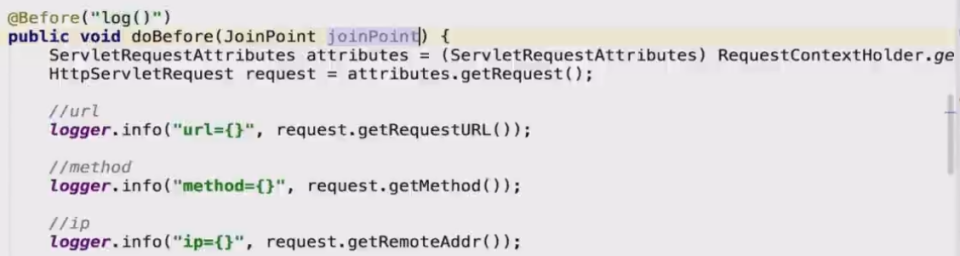
@Component 注入到spring中

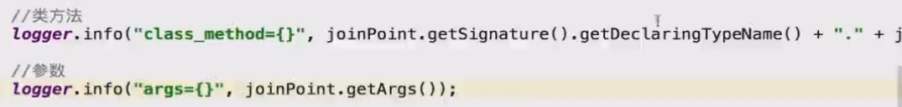
@Before 表示在请求之前

@Pointcut 定义一个公共的方法

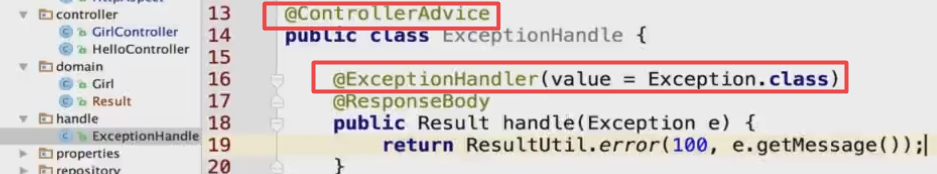
@AfterReturning 获取请求返回值

获取url等值



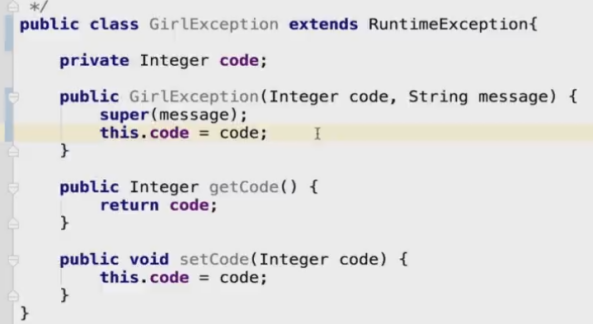


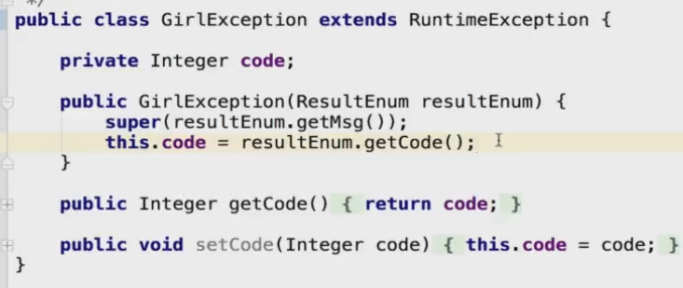
**统一异常处理**

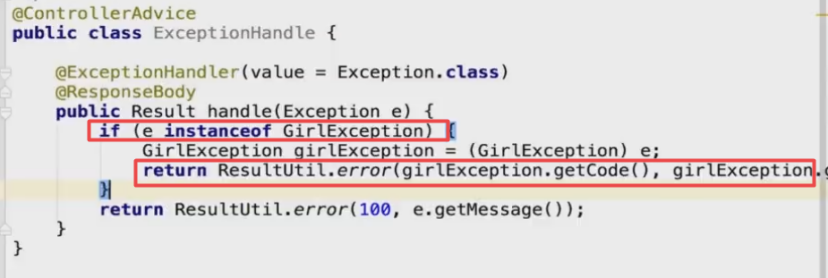


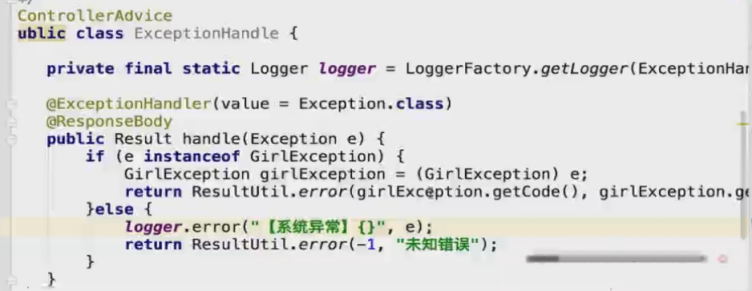
@ExceptionHandler 处理的异常类型

Spring的异常只有继承RuntimeException才会进行回滚，继承Exception不会回滚

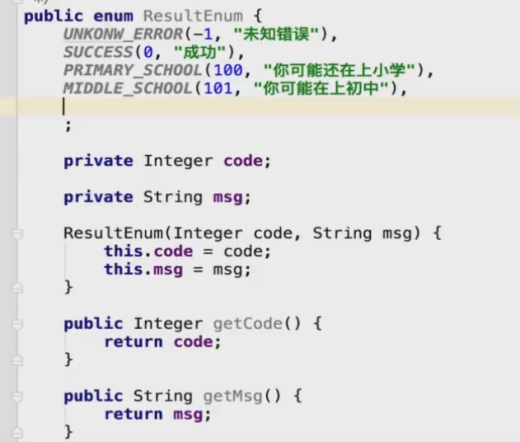


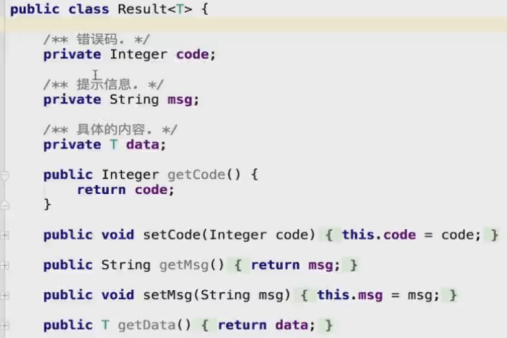






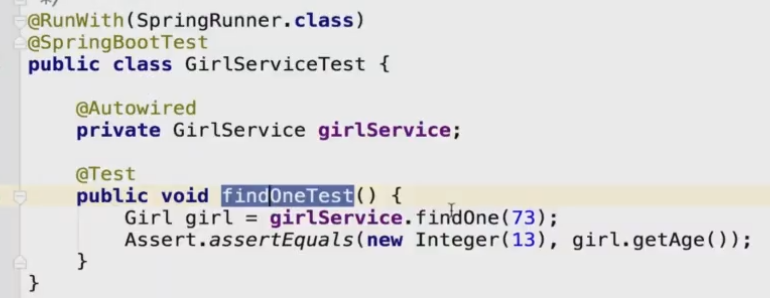
用枚举控制错误码和错误信息(枚举没有set方法)



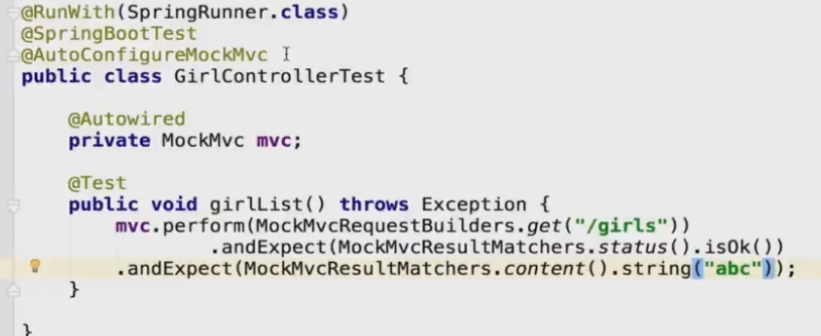


**单元测试**

对service进行单元测试



对controller进行单元测试



Maven进行单元测试并打包



Maven打包的时候跳过单元测试

