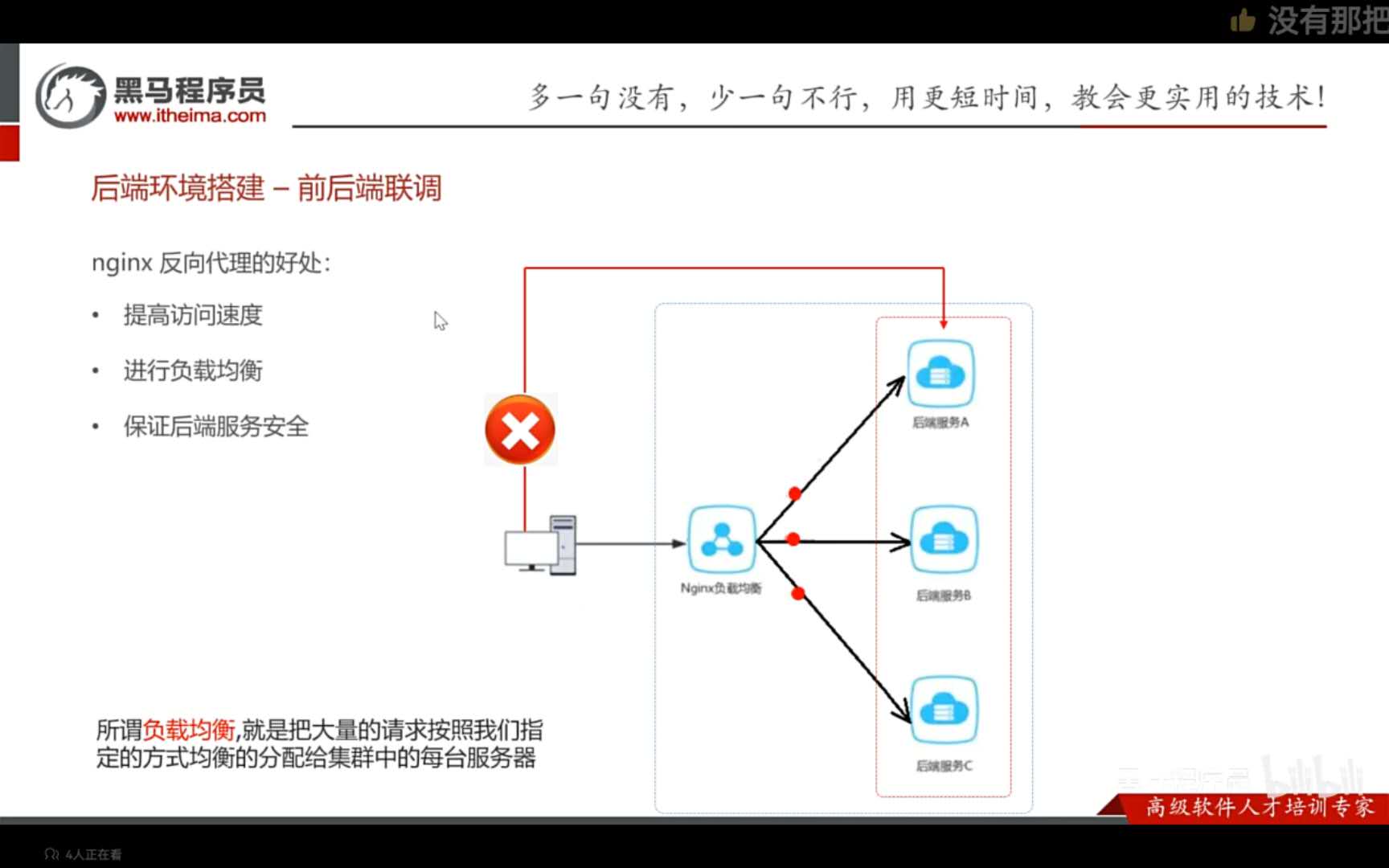
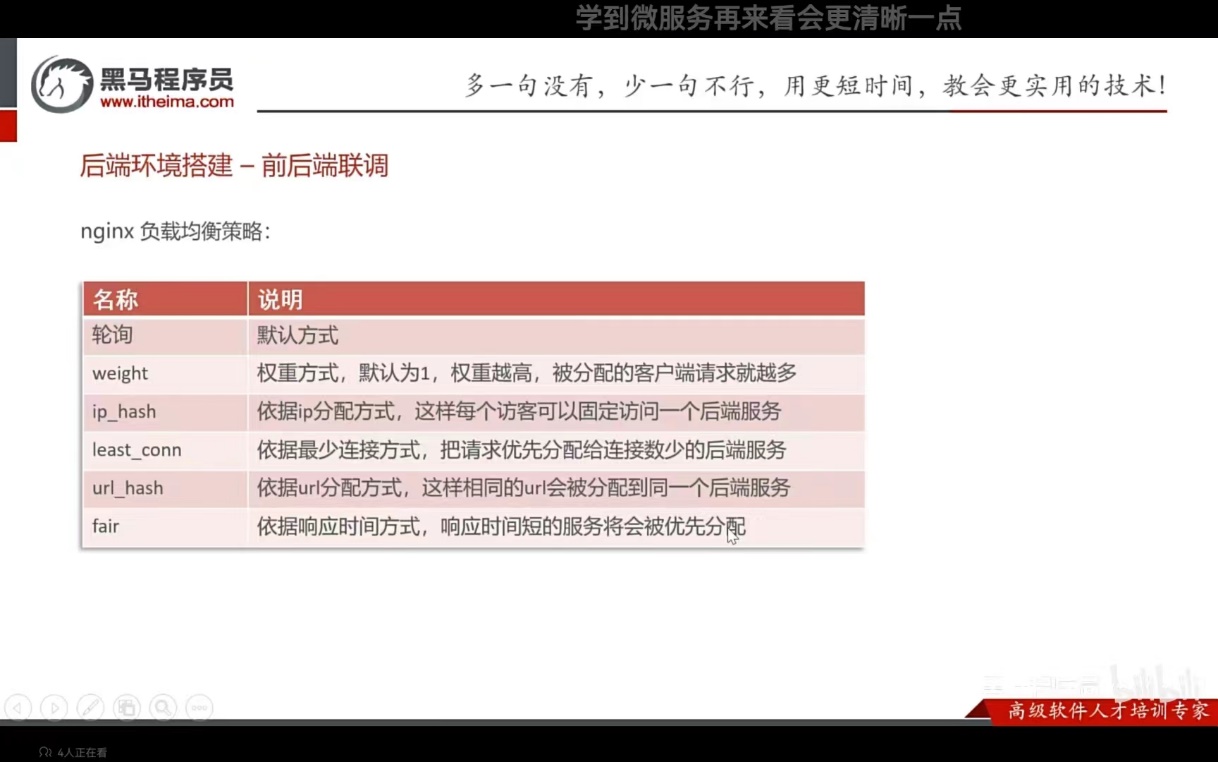
1. Nginx启动错误日志在log目录下
2. Idea创建git环境

VCS: Create Git Repository

1. Idea中sql命令要ctrl+a全选执行。我的数据库版本太低了，是5.1的utf8mb4/utf8mb3需要全文替换成utf8
2. Markdown用VS右键打开侧边预览
3. Ctrl+alt+左 返回
4. 注解模板：输入/\*\*，点击“Enter”

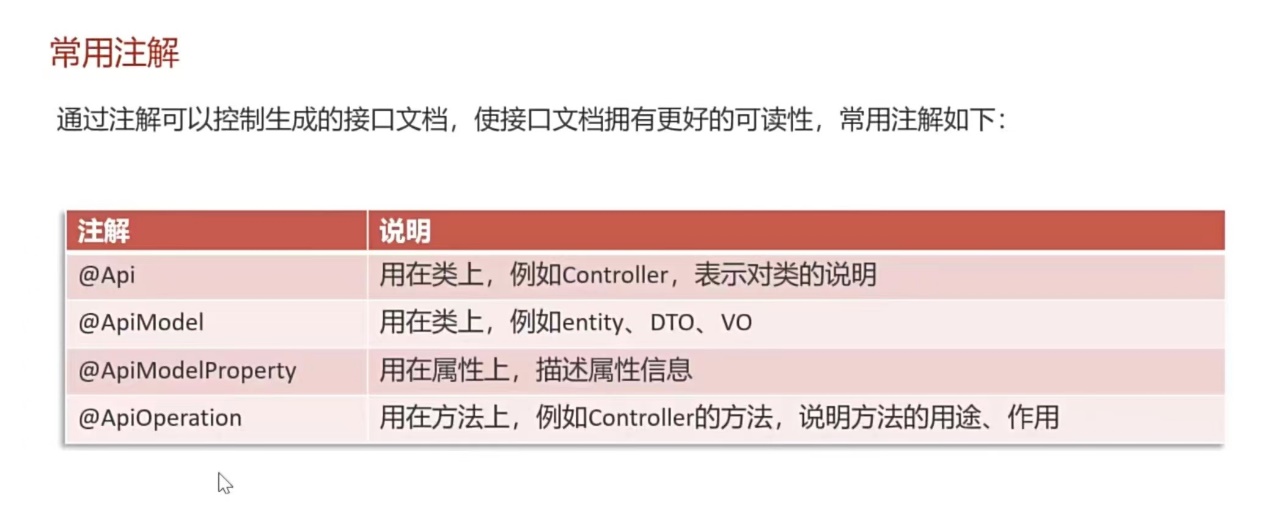


有缓存，可以分发多个后端服务器，不暴露后端内网接口



1. Swagger

WebMvcConfiguration中通过knife4j生成接口文档，可以使用Swagger测试，主要帮助后端开发人员做接口测试



可以设置分组groupName

*/\*\*  
 \* 通过knife4j生成接口文档  
 \** ***@return*** *\*/*@Bean  
public Docket docket1() {  
 ApiInfo apiInfo = new ApiInfoBuilder()  
 .title("苍穹外卖项目接口文档")  
 .version("2.0")  
 .description("苍穹外卖项目接口文档")  
 .build();  
 Docket docket = new Docket(DocumentationType.*SWAGGER\_2*)  
 .groupName("管理端接口")  
 .apiInfo(apiInfo)  
 .select()  
 .apis(RequestHandlerSelectors.*basePackage*("com.sky.controller.admin"))  
 //扫描controller包,路径不要写错  
 .paths(PathSelectors.*any*())  
 .build();  
 return docket;  
}  
  
@Bean  
public Docket docket2() {  
 ApiInfo apiInfo = new ApiInfoBuilder()  
 .title("苍穹外卖项目接口文档")  
 .version("2.0")  
 .description("苍穹外卖项目接口文档")  
 .build();  
 Docket docket = new Docket(DocumentationType.*SWAGGER\_2*)  
 .groupName("用户端接口")  
 .apiInfo(apiInfo)  
 .select()  
 .apis(RequestHandlerSelectors.*basePackage*("com.sky.controller.user"))  
 //扫描controller包,路径不要写错  
 .paths(PathSelectors.*any*())  
 .build();  
 return docket;  
}

1. 实体类添加@Builder就可以通过Builder方法创建对象
2. //对象属性拷贝

BeanUtils.copyProperties(employeeDTO,employee);

//从前面拷贝到后面

PageHelper底层是靠ThreadLocal记录 页码和每页记录数

1. AOP



1. @Transactional

保证原子操作

需要在启动类上

@EnableTransactionManagement //开启注解方式的事务管理

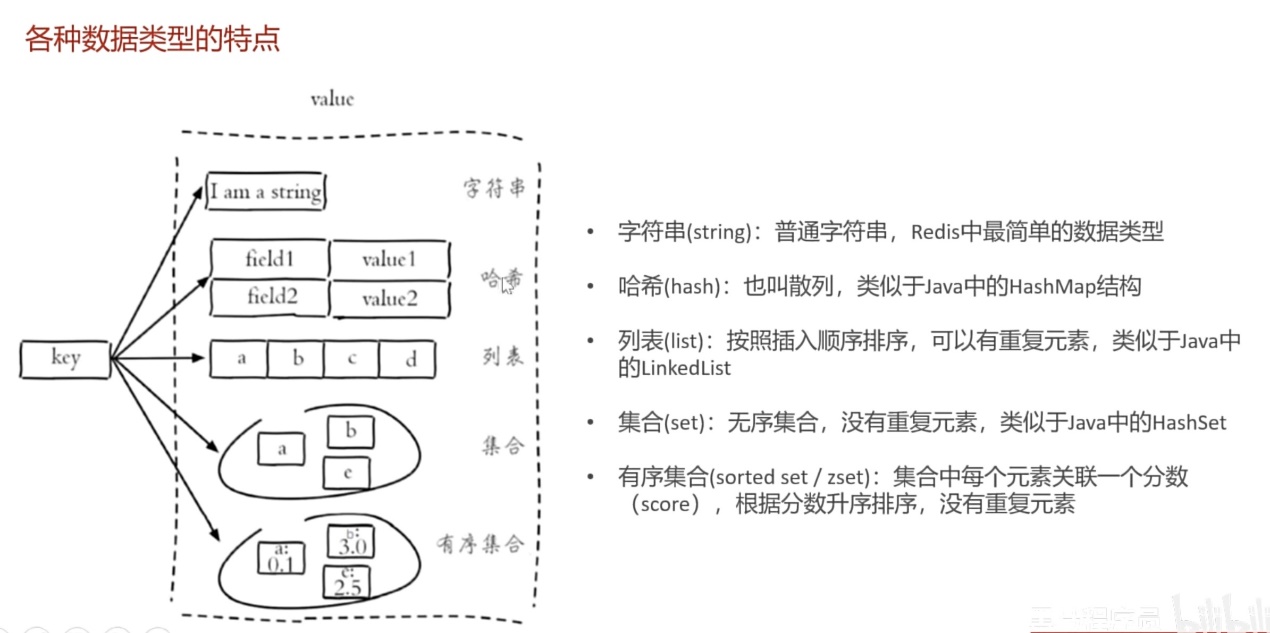
13获取插入后的主键值id

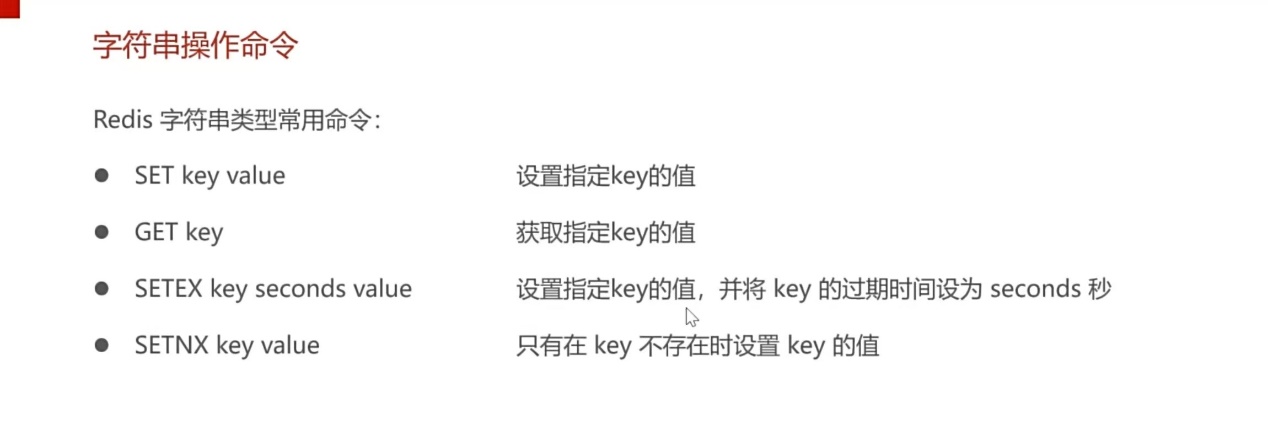
<insert id="insert" useGeneratedKeys="true" keyProperty="id">

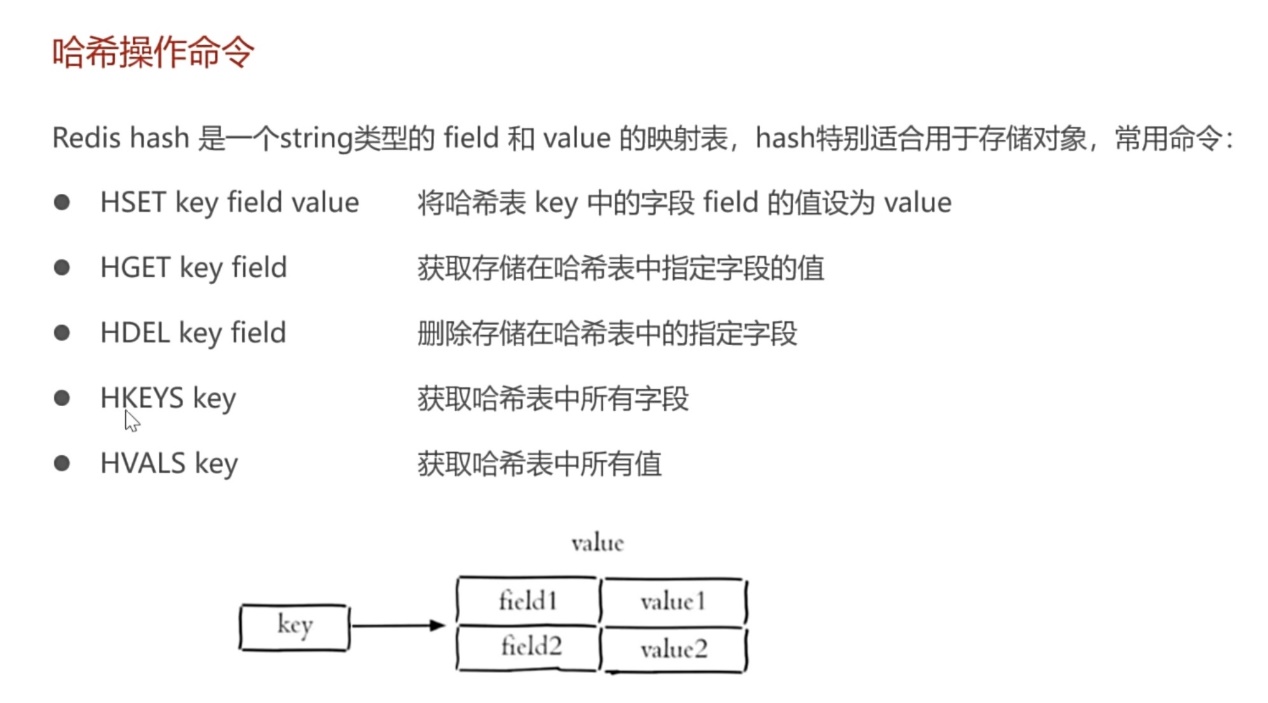
1. 将传递的参数1，2，3字符串自动解析成List集和

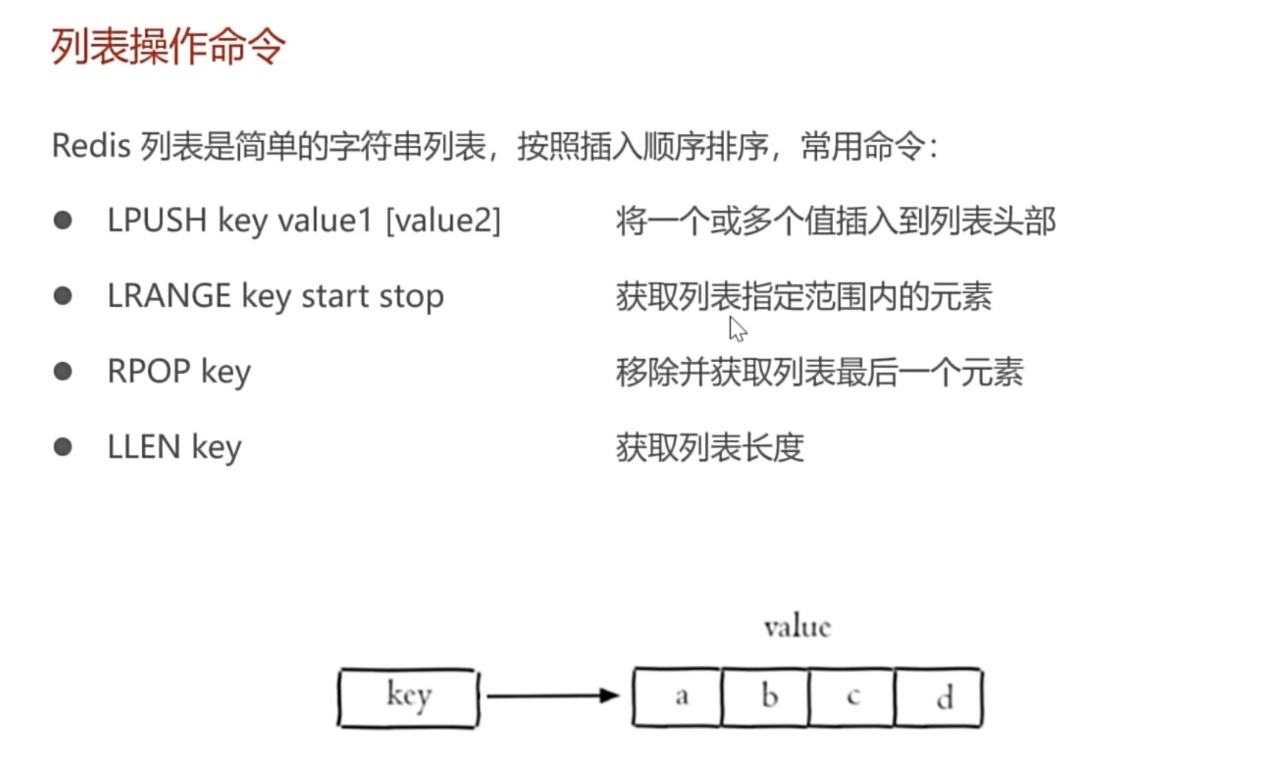
@RequestParam List<Long> ids

1. Redis

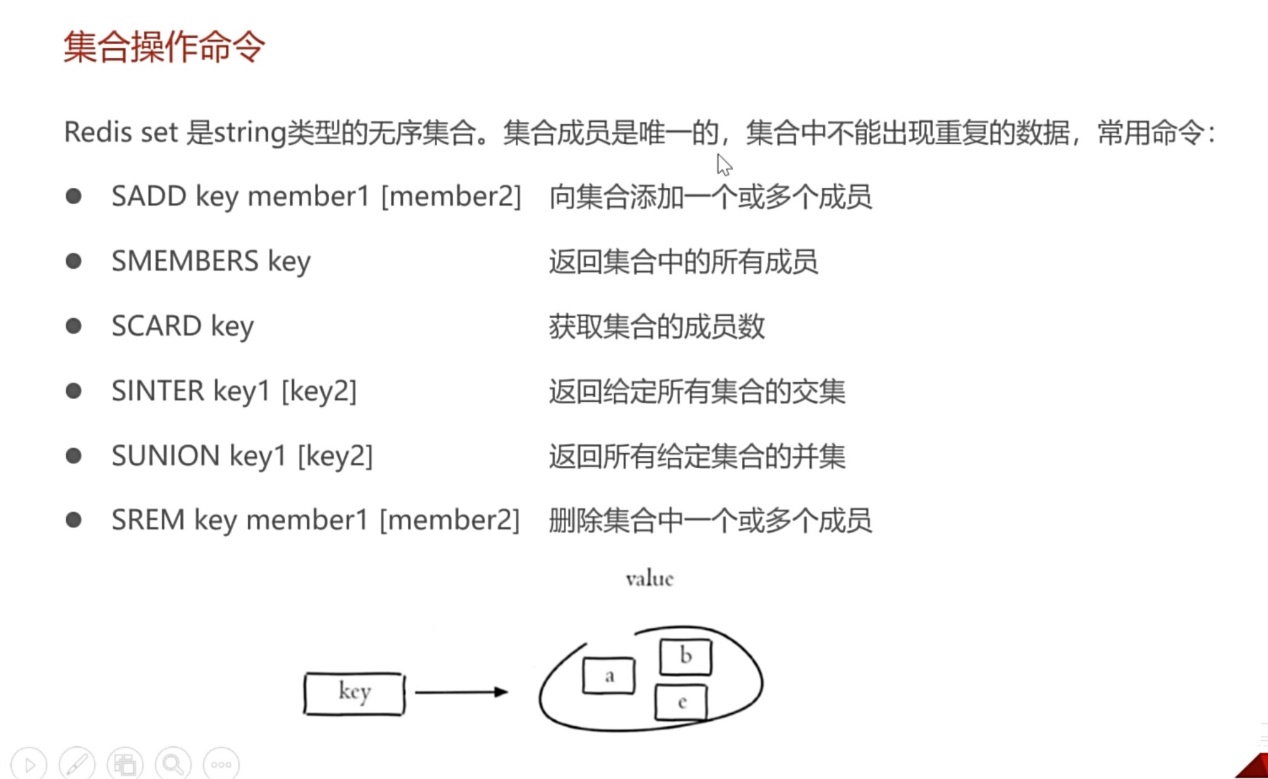


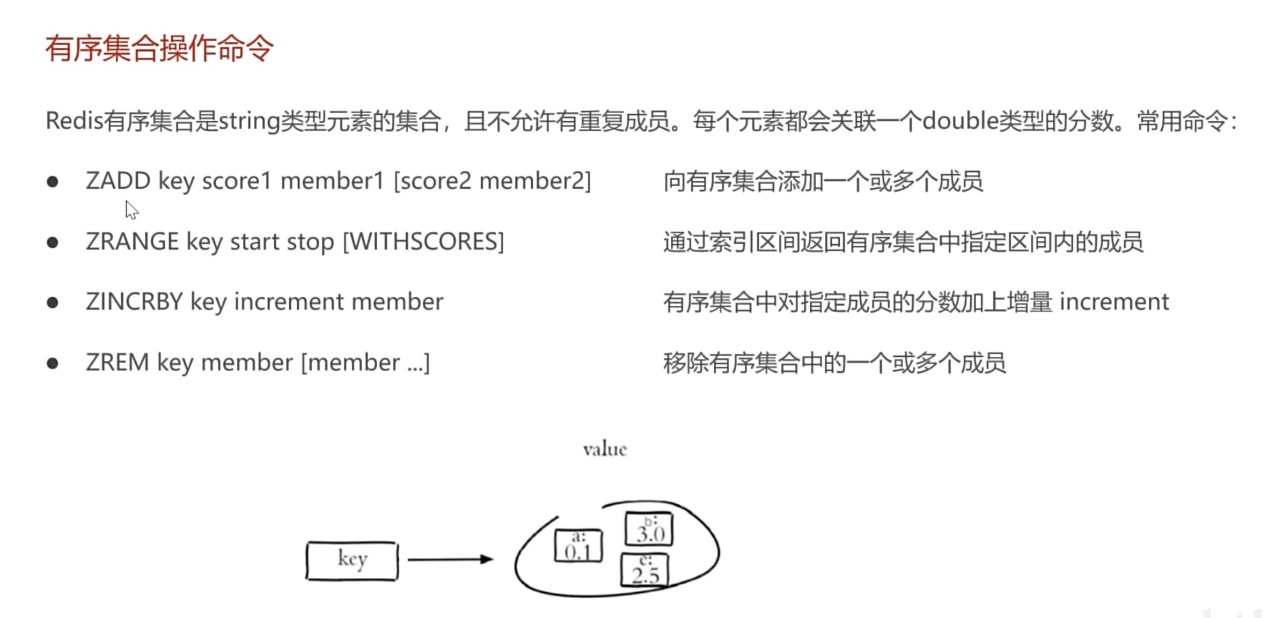






(队列)

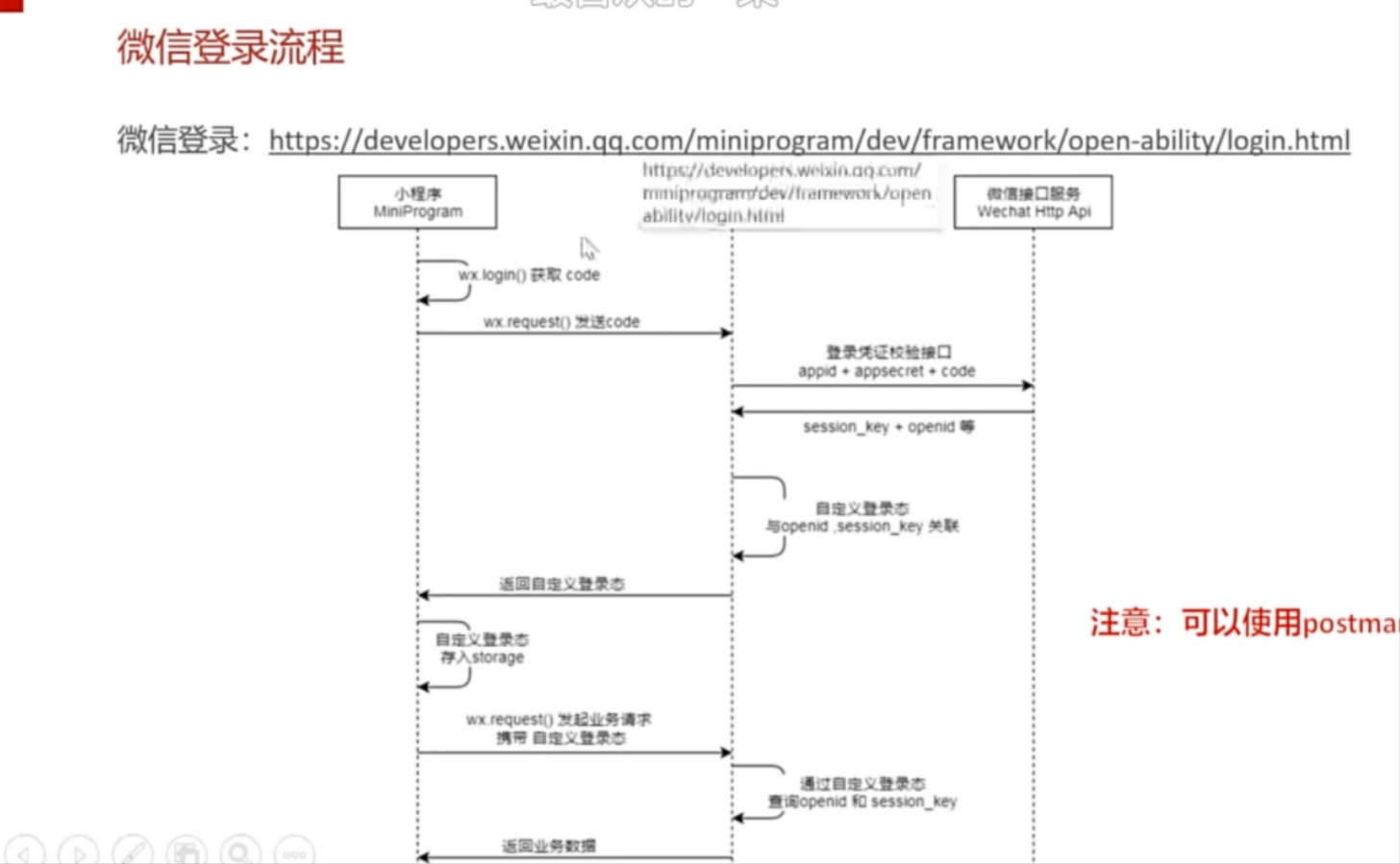




1. 自定义Bean的名称2

@RestController("adminShopController")

1. 微信登录流程



1. **条件查询给他封装到一个对象里，在controller层就封装好，以便后面的接口统一**

Setmeal setmeal = new Setmeal();  
setmeal.setCategoryId(categoryId);  
setmeal.setStatus(StatusConstant.*ENABLE*);  
  
List<Setmeal> list = setmealService.list(setmeal);

1. **SpringCache**

<dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-cache</artifactId>  
</dependency>

@CachePut(cacheNames = "userCache",key="#user.id") //如果使用springCache缓存数据，key的生成：从参数获取  
//先插入数据，再缓存数据  
//@CachePut(cacheNames = "userCache",key="#result.id") //从return返回值获取

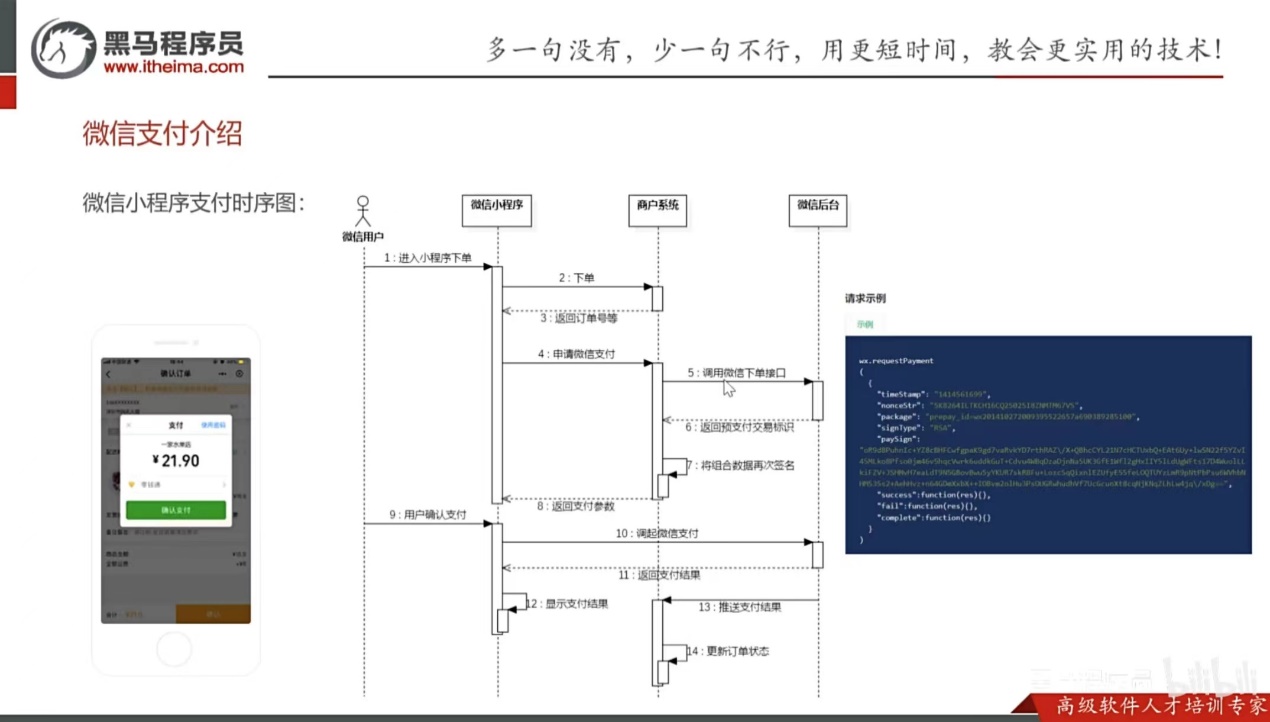
@Cacheable(cacheNames = "userCache",key = "#id")//key:userCache::2  
//如果没查到，会将方法的返回结果缓存

@CacheEvict(cacheNames = "userCache",key = "#id")//清除特定key缓存

@CacheEvict(cacheNames = "userCache",allEntries = true)//清除全部key缓存

1. **快捷键**

* Fn 键+方向左键 = Home 键，
* Fn 键+方向上键 = PgUp 键，
* Fn 键+方向下键 = PgDn 键，
* Fn 键+方向右键 = End 键。

****

1. **又是个版本的bug**

**由于我的数据库版本较低，MySQL驱动选的是5.1.47的**

**在mybatis查询返回订单数据时Debug跳到了一个不熟悉的类里处理异常**

**报了个Error attempting to get column xxx from result set. Cause: java.sql.SQLFe**

**某个列的数据没拿到，查了一下说是日期无法正常序列化与反序列化，看了一下自己配置的日期序列化和反序列化器，肯定没有问题；又有说数据库表类型和实体类型不一致，看一下也是一致的。**

**然后感到很奇怪，往下看另一个错误说**

**Cannot invoke "java.sql.Timestamp.toLocalDateTime()" because the return value of "com.mysql.jdbc.JDBC42ResultSet.getTimestamp(int)" is null**

**居然是驱动包自身bug**

**dbutils中在填充参数时做了一个操作，使用了prepareStatement.getParameterMetaData()方法来获取传入参数在数据库中对应的元数据信息。而返回的ParameterMetaData对象中的metadata.fields属性是null，但是在5.1.47的驱动包中并没有做空值检查，所以就炸了**

**当我们使用dbutils往MySQL中插入TimeStamp类型出现空指针异常，而你确定空指针不是由于你的外部代码导致时，果断看看你MySQL驱动包的版本是不是用的坑爹的5.1.47**

1. **Spring Task**

**定时任务框架：定时自动执行某段Java代码**

**用 cron 表达式在线生成器**

**https://cron.qqe2.com/**

****

1. **WebSocket**

**基于TCP的网络协议，实现浏览器和服务器全双工通信，创建持久连接，进行数据双向传输**

**HTTP：单向的，（短连接？）**

**WebSocket：双向的，长连接**

**WebSocket应用：（网页未刷新，数据实时更新）**

1. **视频弹幕**
2. **网页聊天**
3. **股票基金实时报价**

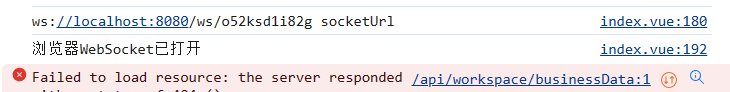
**快速终止nginx进程**

taskkill /F /IM nginx.exe > nul

**在开发中遇到了奇怪bug，ws://localhost/ws/xxxx没请求到，去前端源码改了**

**ws://localhost:8080/ws/xxxx才成功。不知道为什么。**

**改了ngnix中的代码在edge请求路径仍然没变，清理缓存后成功了， edge浏览器要清缓存。**



1. **Stream流**

List<GoodsSalesDTO> salesTop10 = orderMapper.getSalesTop10(beginTime, endTime);  
List<String> names = salesTop10.stream().map(GoodsSalesDTO::getName).collect(Collectors.*toList*());  
String nameList = StringUtils.*join*(names, ",");  
List<Integer> numbers = salesTop10.stream().map(GoodsSalesDTO::getNumber).collect(Collectors.*toList*());  
String numberList = StringUtils.*join*(numbers, ",");

1. **Apache POI**

**导出Excel报表数据**