## 关于对模糊搜索功能的总结

在经过前后2天的反复尝试与分析，终于把模糊搜索功能解决了。鸡冻啊！！

#### 一、问题

其中最大的问题莫过于在用户搜索时如何处理用户误操作输入，此次处理的是空格字符问题。既要考虑输入是否为空，也要考虑非空情况下是否有值，加上是两个对象，情况变得相对复杂。其次问题也在mybatis的xml配置书写，一是对于like后的格式也查了很多资料，最终落实；二是如何根据不同的情况书写查询语句。

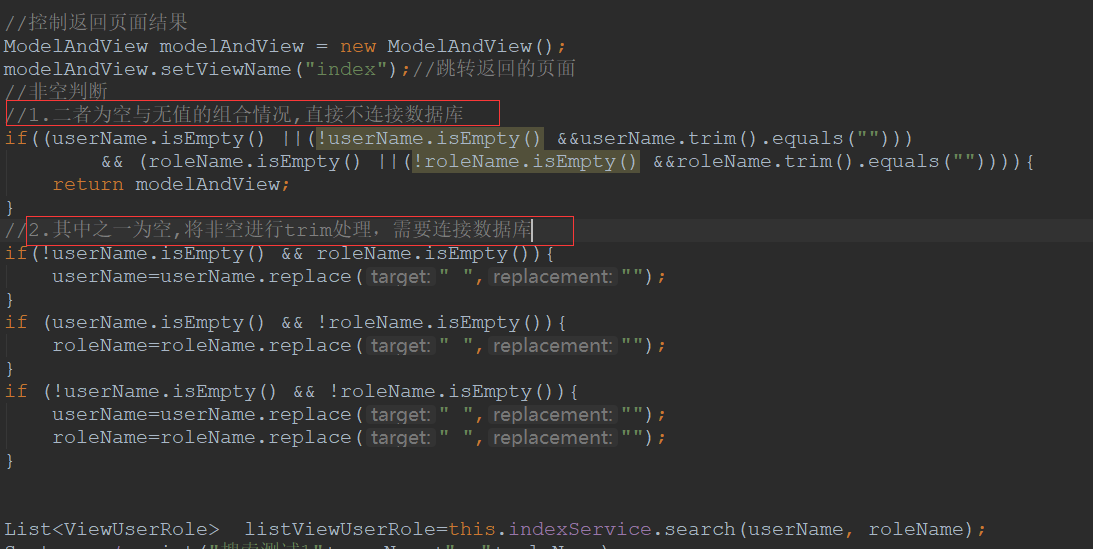
#### 二、设计流程（前端到后端，此处没有写service层）

（一）对用户输入的各种情况进行分析归类并进行清除空格处理（如果不处理，到mybatis层会出现问题）

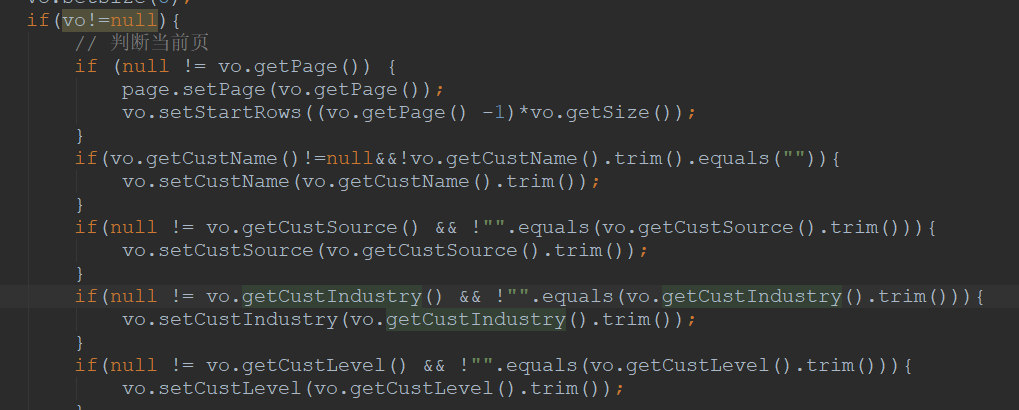
1.知识点

replace(" ","");函数，将字符串的空格全部清除，包括前中后

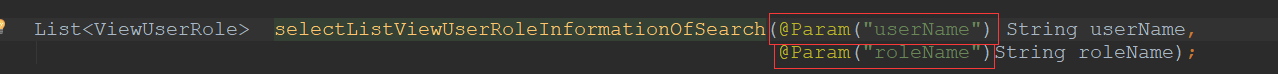
trim()只是处理前后的空格



2.优化——将前台对象封装成QueryVo，更好操作



（二）mapper的书写（多参数时需要用@Param()注解）



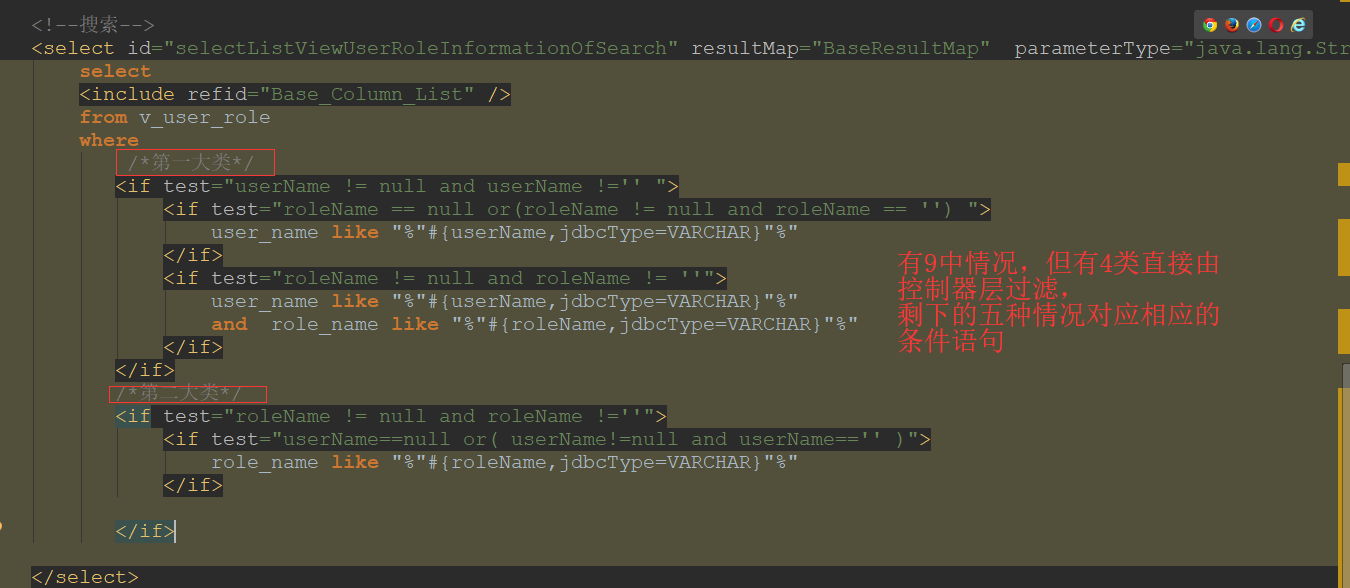
（三）xml文件的配置书写

1.like语句的书写

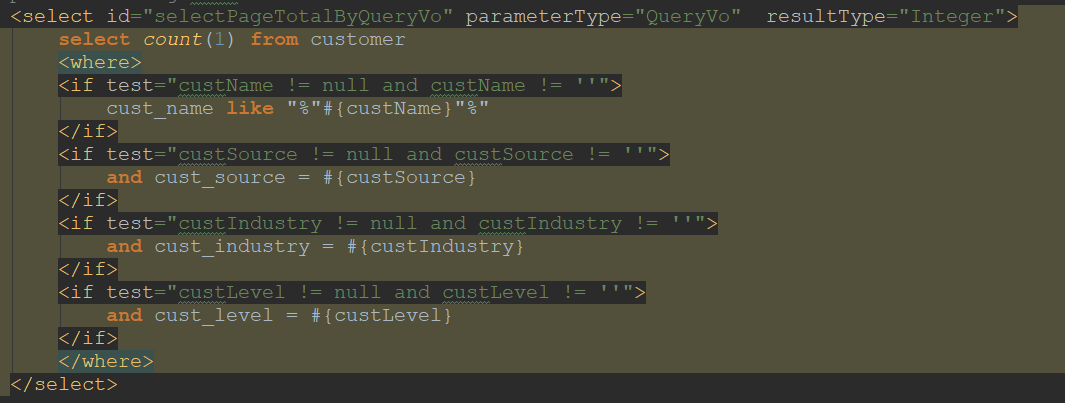
<if test="userName==null or( userName!=null and userName=='' )">

role\_name like **"%"#{roleName,jdbcType=VARCHAR}"%"**

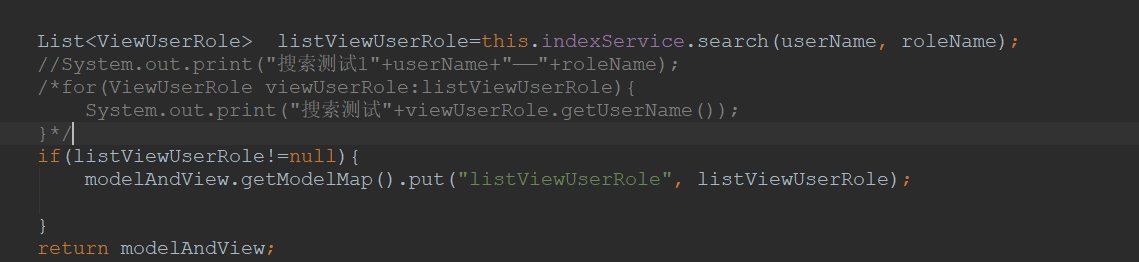
</if>



2.对sql语句的优化——使用<where></where>标签，自动去除前and



（四）对传回的数据进行处理



#### 三、体会

（1）一个合理化的设计流程是多么的重要，在本次设计中深有体会。

在简单的逻辑与分类不太复杂的情况下，预先设计设计流程似乎显得有些多余，可是在进入到复杂的项目与团队协作时又是那么的不可缺少，而这正是自己所缺少的。

（2）另外，自己在考虑问题是不要怕复杂，这样容易考虑不全。

本次又9中情况，可进行3大类区分，由于没有仔细分析，导致在控制层的逻辑处理与mybatis的xml语句配置都出现大的问题。

此处，使用<where>便可简化操作，书写简单不易错。So,系统完整的学习十分重要