**北 京 林 业 大 学**

**2020 学年— 2021学年第一学期 JavaEE实验报告书**

专 业：计算机科学与技术 班 级： 18-2班

姓 名： 黄国亮 学 号： 181002212

实验地点： 计算中心 任课教师： 王新阳

实验题目： **基于Spring MVC 的MVC架构开发**

实验环境： MyEclipse For JavaEE, Tomcat, JRE

**实验方式：先到相应的教学幻灯片中找到相应的教学内容和示例程序，先理解透框架结构的整体思路和流程，再根据实例代码编写程序。**

实验内容：

**实验任务一（给循序渐进学习的同学）：**

1）在Eclipse下新建一个动态web项目，该项目以“**7-SpringMVC-学号**”为名。

2) 实现一个基于SpringMVC+Spring+MyBatis的留言系统。

实验过程：

1. 引入jar包。

这里我直接把SpringMVC+Spring+MyBatis所需要的包全部导入进来了，如下图所示：

看着太多了，我就不放图了，得好几页。

1. 创建web.xml并部署DispatcherServlet。

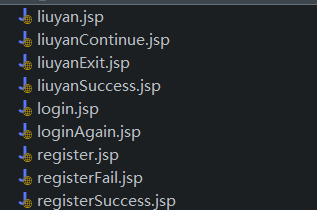
这想相当于告知servlet对象springMVC初始化时，寻找哪个配置文件的问题。

我这里用的配置文件是applicationContext.xml，如下图所示：



1. 创建web页面

这里主要做的是一个留言系统，所以我就简单地做了一个登录页面（login.jsp）,登录失败页面（loginAgain）,登录成功进入留言页面（liuyan.jsp）,留言成功页面（liuyanSuccess.jsp，做一次选择，是否继续留言），继续留言页面（liuyanContinue.jsp），不继续留言页面（也就是退出留言系统页面（liuyanExit.jsp）），注册页面（register.jsp），注册成功页面（registerSuccess.jsp），注册失败页面（registerFail.jsp）。

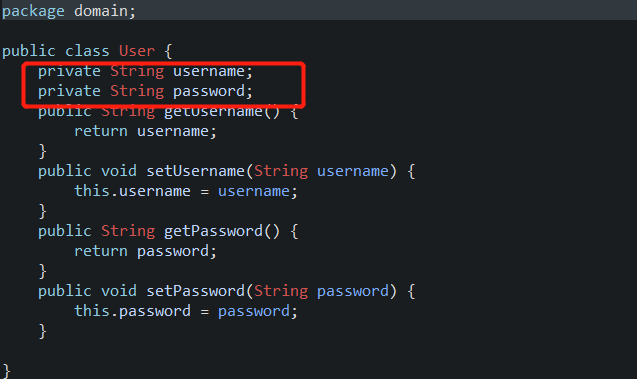


1. 创建持久类domain

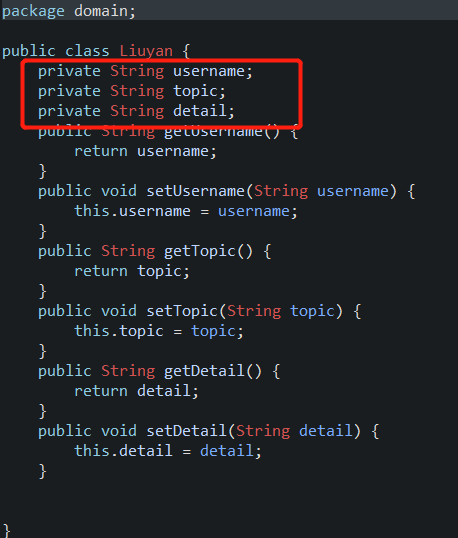
这里因为只设计到用户信息和留言信息，所以只需要创建两个实体类，分别为User.java和Liuyan.java，如下图所示：



User.java包含用户名username和密码password两个属性：



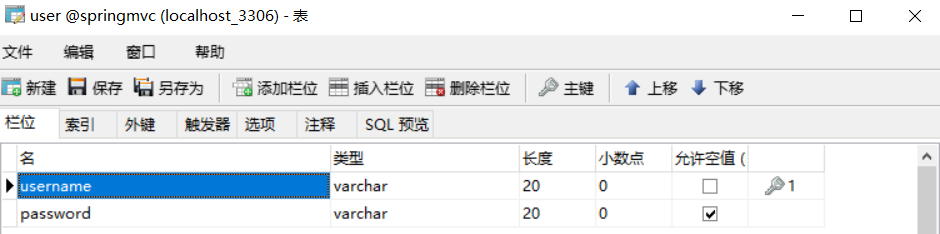
Liuyan.java有三个属性，分别为用户名username，留言主题topic，留言内容detail，如下图所示：



1. 数据库的创建

在将dao层具体操作之前把数据库的创建和连接都先讲了吧，这里我创建的是springmvc数据库，里面创建了两个表，user表和liuyan表，分别用于管理用户信息和留言信息，其中user表中的属性有username为主码，password；而liuyan表中有三个属性，分别是username且为user表中username的外码，topic用于记录留言主题，detail用于记录留言内容，如下图所示：







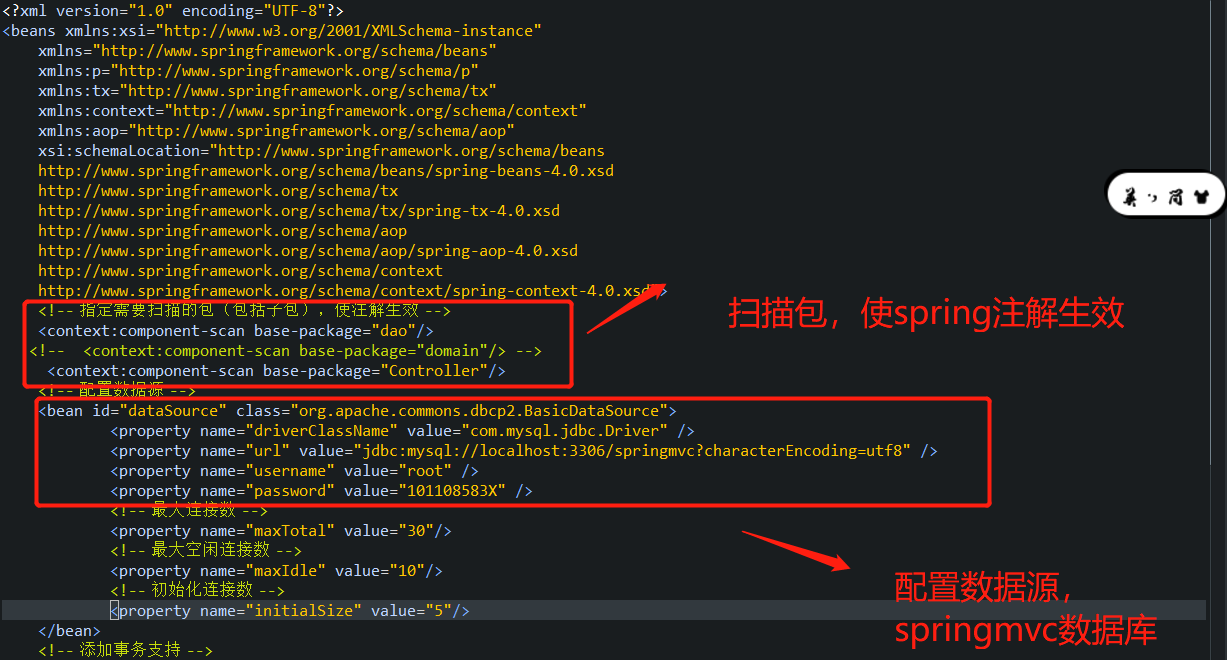


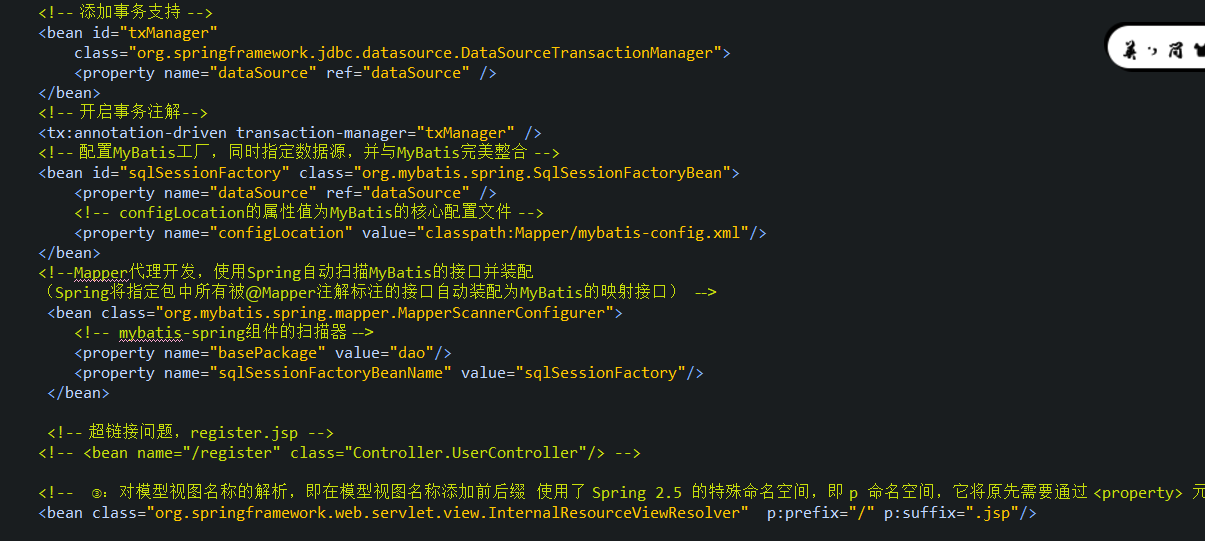
1. 用mybatis连接数据库，和进行一些spring的配置

首先引入spring的各类框架文件，再进行包的扫描，主要是通过扫描包使spring注解生效，再下面配置mybatis数据源，数据库名称为springmvc。

设置一些数据库的最大连接数什么的，相当于设置一个数据库连接池。后面就添加tx事务支持和事务注释，并通过设置spring工厂bean并用mapper代理开发，使用spring时自动扫描mybatis接口并进行整合，这样就能让spring将指定包中所有被@mapper注释标注的接口自动装配为mybatis的映射接口。

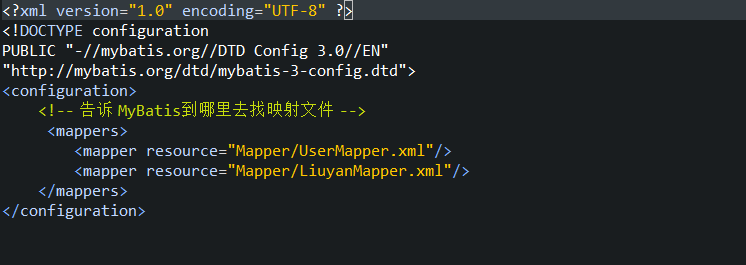
最后对模型视图名称的解析，即在模型视图名称添加前后缀使用了 Spring 2.5 的特殊命名空间，即 p 命名空间，它将原先需要通过 <property> 元素配置的内容转化为 <bean> 属性配置，在一定程度上简化了 <bean> 的配置。



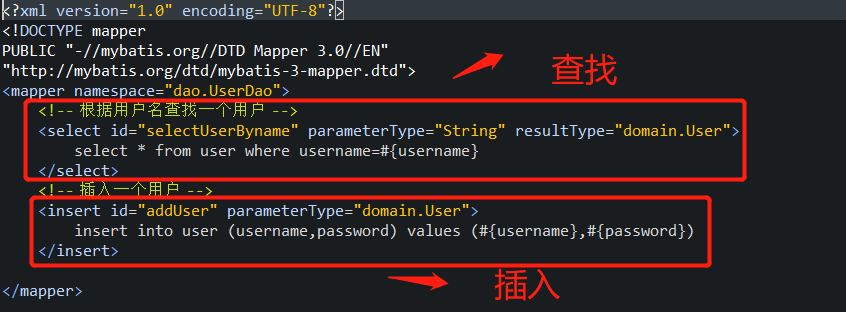


1. 映射文件的配置

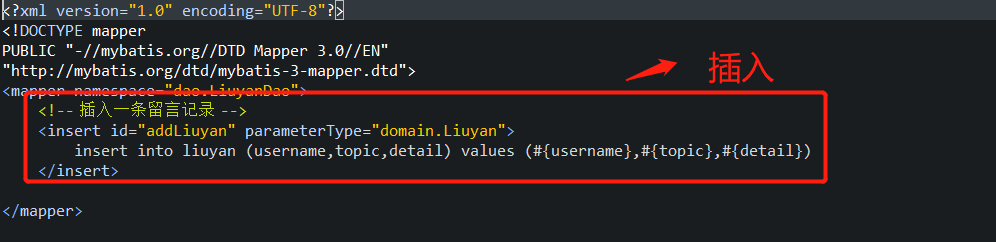
Mybatis-config.xml文件，这个文件主要用来管理映射文件，相当于mybatis工厂，这里有两个映射文件，分别是UserMapper.xml，LiuyanMapper.xml，如下图所示：



UserMapper.xml文件：



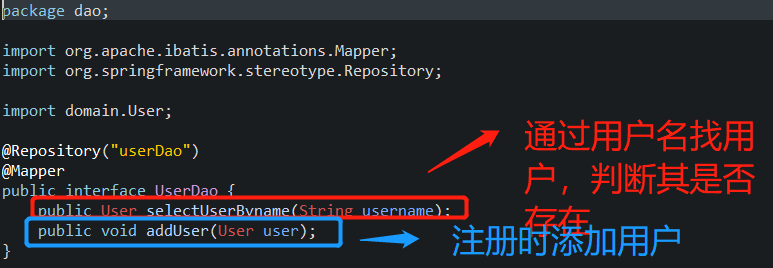
LiuyanMapper.xml



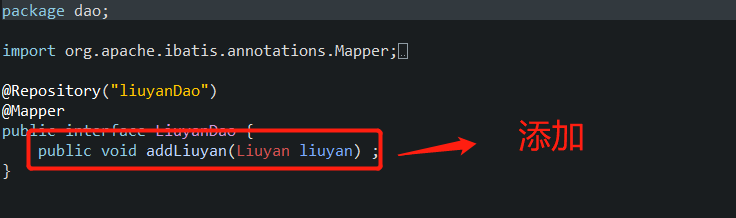
1. dao层创建

同样这里也只有两个接口，分别是UserDao和LiuyanDao，分别将对数据库中user表和liuyan表的操作封装起来，这些方法名和xml映射文件的id是相等对应的，如下图所示：

user中因为只需要两个操作，第一个通过用户名遍历数据库判断用户是否存在，第二个操作就是注册时需要的向数据库添加用户，如下图所示：

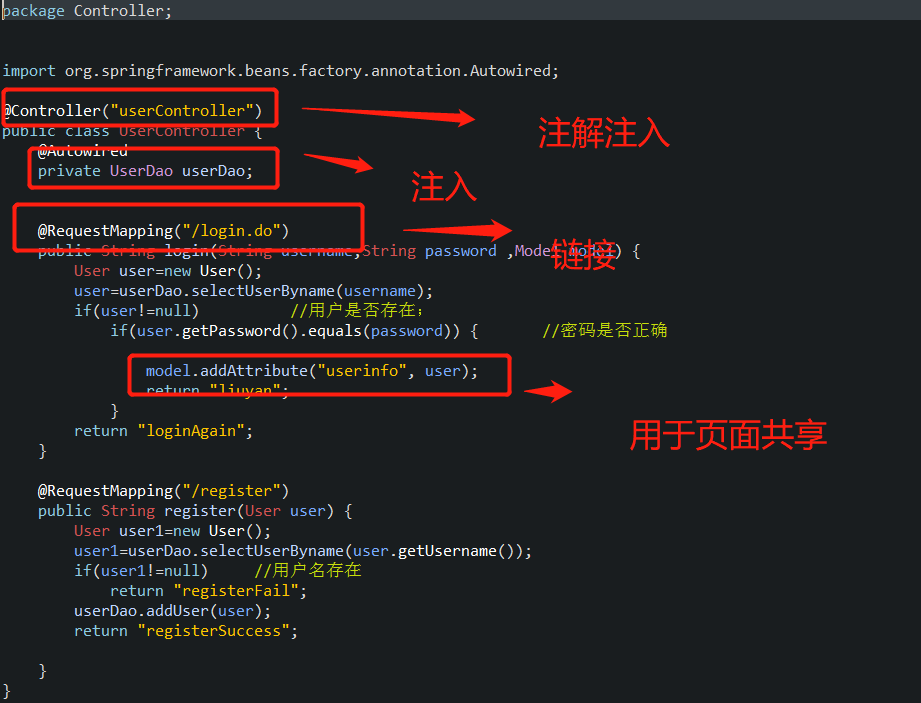


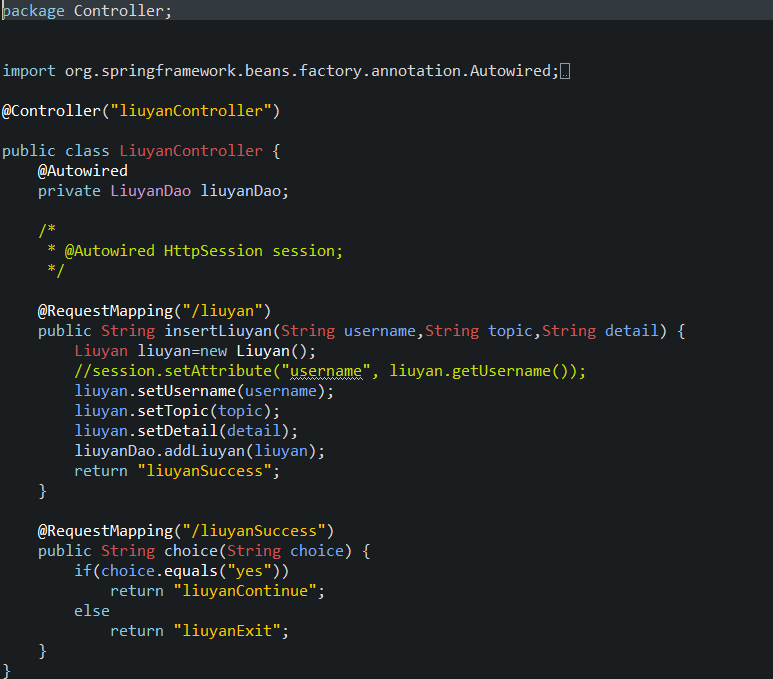
liuyan只有一个操作，也就是进行留言，相当于向数据库中添加留言信息，所以如下图所示：



1. 创建Controller类

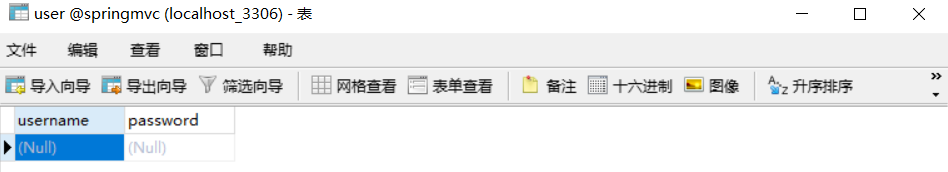
同样这里也有两个Controller类，一个userController控制类，一个liuyanController控制类，用于控制页面跳转和页面共享，如下图所示：

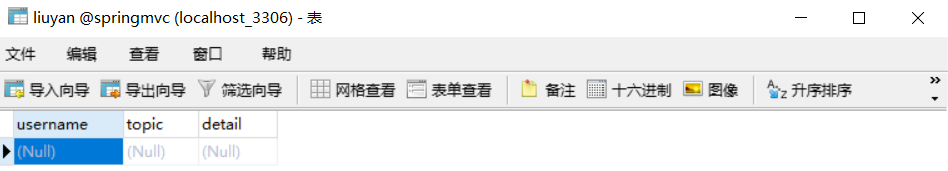




1. login.jsp登录界面

现在开始演示吧！一些内容边演示边解释！我先把数据库里的内容全部清空，所以刚开始没有user用户，所以先看看吧：

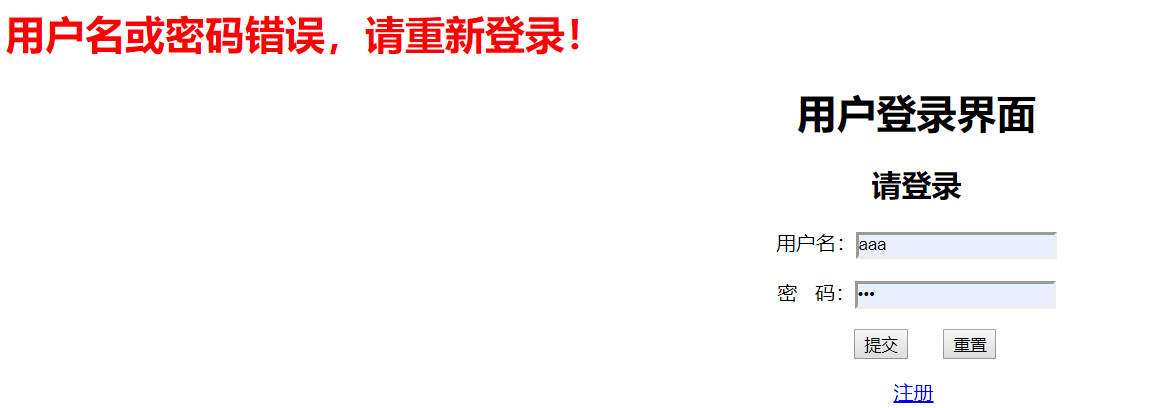








没有aaa用户所以会显示用户名或密码错误，重新登录。

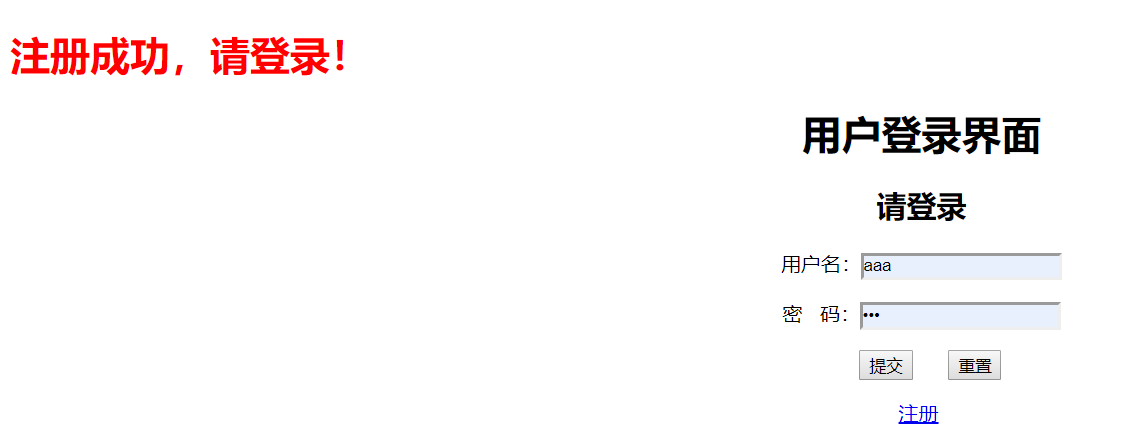


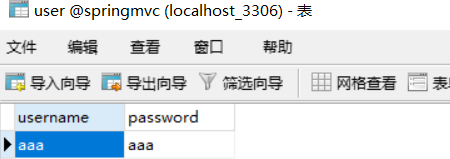
1. register.jsp

因为现在没有用户，所以我先注册吧！！

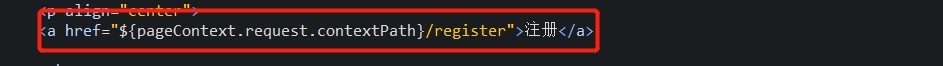








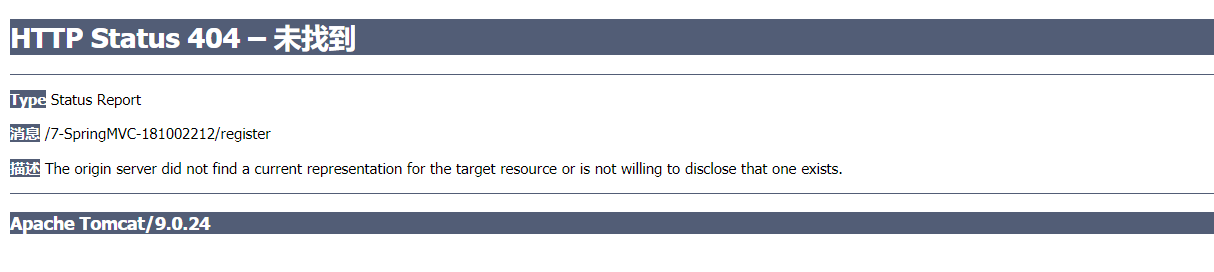
但是！！！有一个问题因为我之前login.jsp里写的是这个，所以当你不是在登录界面点击注册时会报错！！！看下图：



这个相当于在registerSuccess.jsp页面，只是获取了login.jsp页面的内容，所以我们点击注册会出现一个神奇的现象！！请看：



找不到！！这就非常奇怪了！！难度这是因为不在同一个request内嘛？？我不是很理解。



1. liuyan.jsp

好了，刚刚已经创建了用户名为aaa的用户，所以我们可以登录了，看下图：



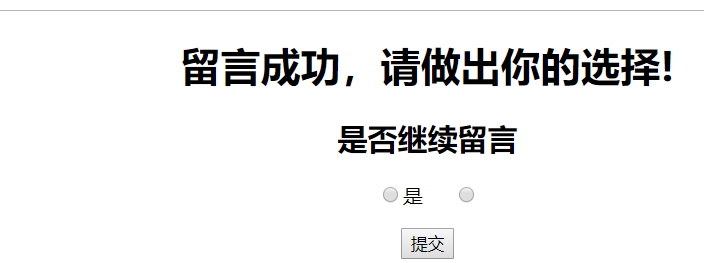
进入了留言页面：

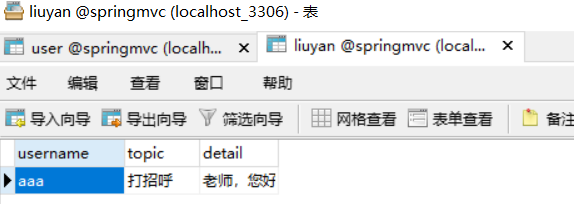


1. liuyanSuccess.jsp

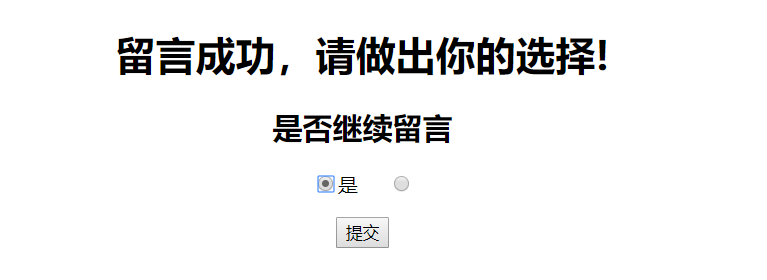
点击提交后，表单页面内容存入数据库中，并跳转到liuyanSuccess.jsp页面，如下图所示：



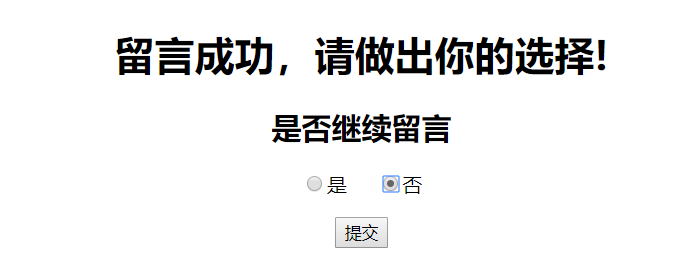




这里我做了一个选择，如果选择是的话，则跳转到liuyanContinue页面，选择否的话则退出留言系统到liuyanExit.jsp页面，这里我就不细讲，看图片：

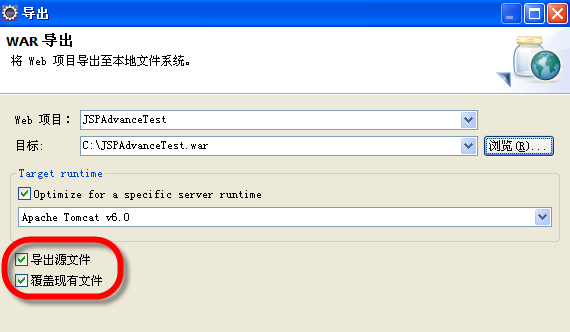








实验要求：

1. **利用SpringMVC+Spring+MyBatis实现一个简单的登录表单提交功能，并实现用户名和密码的合法性验证等业务逻辑。登录后页面跳转至留言页面，实现留言功能，留言页面至少要包含留言的主题、内容等信息，用户点击留言提交按钮后，通过SpringMVC实现相关业务逻辑的控制，并由MyBatis框架实现留言相关信息的持久化。**
2. **导出项目，导出时注意选择包含源文件，如下图所示  
   **
3. **上述War文件、实验报告文件以及源程序代码一起打包，命名为“7-学号-姓名”提交至**[**ftp://202.204.121.124/王新阳/**](ftp://202.204.121.124/王新阳/)**课程作业/相应班级目录下。提交截止时间为期末考试前。**
4. 有解决思路的介绍说明和文字描述。
5. 请在此把程序运行结果抓图下来。