**JAVA大作业报告**

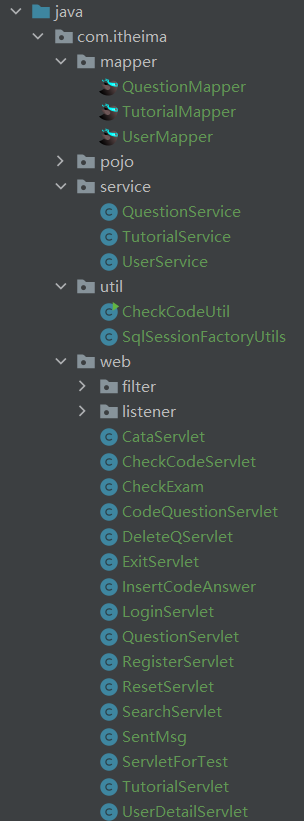
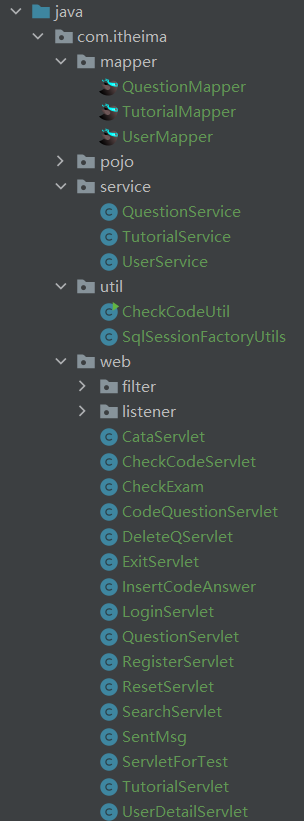
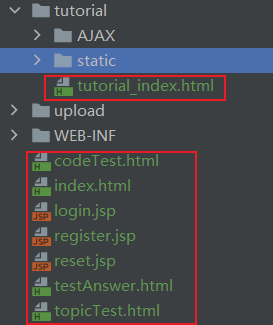
1. **工作描述**
   1. **需求分析与分工**

我出于兴趣在暑假开始学习javaweb的内容，对java建站有一定的了解。在前期分工的过程中，主要由我来对网站设计的需求进行分析，并把工作拆分成两份，并向队友传授JavaWeb的学习路径。

* 1. **习题库和知识库**

编写主页、习题库和知识库的前端页面。

采用MVC框架,编写了众多Controller和Model,实现java后端逻辑处理和数据模型.



* 1. **习题与知识的链接**

实现了习题到知识库的链接，点击知识点超链接，可跳转到包含该知识点最多字段的教程页面。



* 1. **部署到云服务器**

负责网站项目上云，搭建云服务器的环境，安装了宝塔、mysql、tomcat9等工具。反复折腾，才把网页项目上到云上。

购买域名，并进行域名解析，与公网ip进行绑定。

* 1. **登录注册和重置密码**

采用JSP技术，编写登录注册和重置密码的前端页面，实现前端快速接受后端响应，让用户及时得到登录注册结果。

Java后端实现了登录注册功能。增加了图形验证码功能，防止机器注册。增加了手机验证码的功能，方便用户修改密码。自费30￥购买短信验证码云服务，在增加手机验证码功能的同时，学会了如何调用API接口。

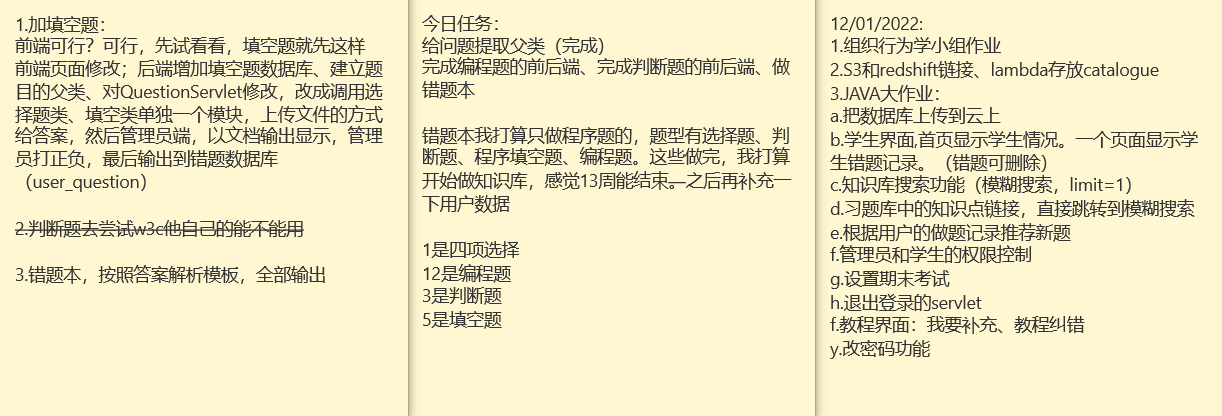
采用了filter技术，实现登录拦截。如果没有通过登录，在Session中留下用户名，无法访问出登陆注册重置密码页面的其他页面。

* 1. **主页的用户学习详情**

根据用户学习和答题情况，在主页显示用户的练习题数



1. **工作需求分析**



* 1. **登录注册**

登陆，需验证账号密码是否正确。记住密码，采用Cookie记录，保存在浏览器端。

注册，需验证账号是否存在，图片验证码和手机验证码是否正确和是否在有效期内。

重置密码与注册类似，把Insert逻辑改成update。

* 1. **习题库和知识库**

习题库，分为单元测试、编程题测试、期末测试。题目包含选择题、判断题、程序填空题，形式多样，内容丰富。除了编程题以外，其他题目在做完时一律自动批改，返回正确答案、知识点、答案解析、正确率、答题时间。习题库可以自动记录答案、答题情况、答题时间等信息，以备题目智能推荐。

知识库，包含父级知识点和子级知识点，点击侧栏即可跳转到对应知识点。知识库还添加了字段搜索功能，可根据输入的字段搜索出你需要的教程。

* 1. **习题与知识的链接**

在完成题目后，我们通常需要深入学习每道题背后的知识点。本项目通过在答案解析出添加知识点的超链接，点击可自动跳转到对应的教程页面。

* 1. **云部署**

购买服务器，安装运行环境，部署之后debug

1. **实现解析**
   1. **习题库和知识库**
      1. 习题库

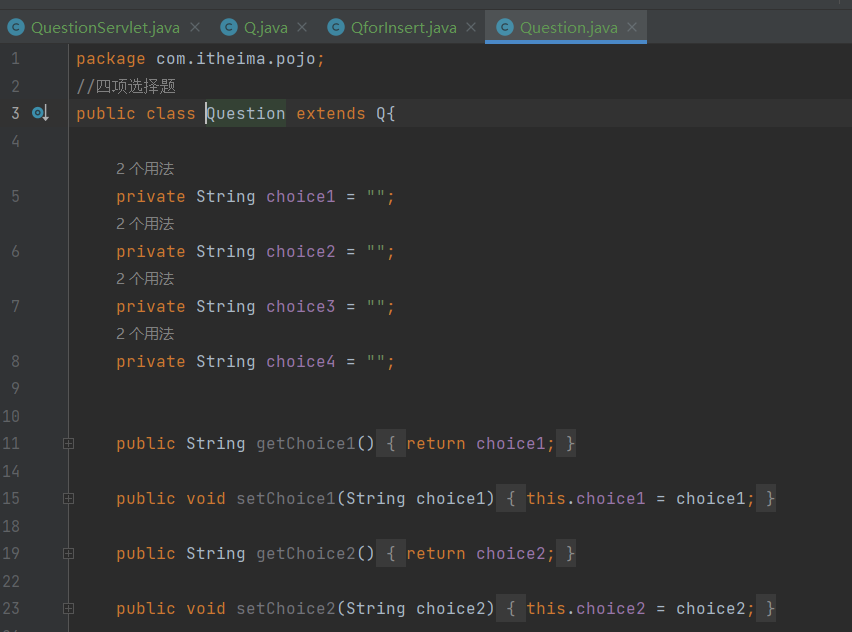
根据主页的url超链接中的参数topic，检索数据库中对应单元的试题。随机抽取几道选择题、判断题和改错题。Topic=0时，为期末考试，搜索所有单元的试题。Topic=1时，为错题集，搜索用户所错的题。在完成错题集后，用户可以删除对应错题。

该部分前端逻辑复杂，需采用AJAX进行局部更新，具体见src/main/webapp/js/topicTest\_AJAX.js。后端采用List集合类，存储试题对象。

试题对象统一父类Q对象，见src/main/java/com/itheima/pojo/Q.java，，选择题、判断题等题目继承Q对象，并定义自己的属性和方法，实现了继承和多态。



所有题目对象中的属性外对象均不能访问，在此提供了get、set方法，对对象的数据进行访问，提供了对外的接口。



答完题交卷后，AJAX访问名为CheckExam的Servlet，处理用户的答案和答题情况，返回给topicTest.html页面。该页面接收到Response之后，将Response储存在LocalStorage中，并跳转到testAnswer.html页面。在testAnswer.html页面中，调取LocalStorage中批改完的信息，局部刷新，显示在页面中。

在错题集的testAnswer.html页面中，还增加了删除题目的按钮，点击后调用src/main/java/com/itheima/web/DeleteQServlet.java，删除用户错题库中的错题，下次刷错题就不会再次出现该题。

* + 1. 知识库



进入该页面时，AJAX调用CataServlet.java和TutorialServlet，加载侧边栏的目录和正文部分。

右上角搜索框，点击按钮或回车后，调用SearchServlet，到数据库中搜索与输入字段内容最相近的教程，封装成对象后，返回给前端页面呈现。

前端页面还增加了字体大小调整、黑夜模式、关闭侧边栏等功能。

* 1. **登录注册与重置密码**
     1. 登录

获取前端返回的username和password，去调取数据库进行验证。成功，把username储存到Session中，防止之后访问其他页面被filter拦截。Filter通过核验Session中的用户名，对用户的操作放行。

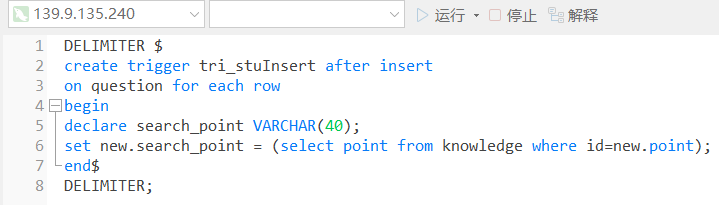
* + 1. 注册与重置密码

注册逻辑较为复杂。首先根据用户名判断用户是否已存在，是，则在页面中提醒用户。第二，采用CheckCodeUtil工具类，生成验证码，存储在Session中，防止机器人注册。点击验证码可更换验证码。注册时，仍会核验验证码是否正确。第三，SentMsg的Servlet实现了短信发送，仍采用Session保存电话号码、验证码、发送时间，保证用户账号能与手机号码绑定。

注册成功后，跳转到登录页面。

* 1. **习题和知识库的链接**

主要采用了数据库的触发器，当后台系统添加题目时，触发器会根据题目链接的知识点id，去索引对应具体的知识点，如“java 数据结构”。在答案解析中显示，并添加超链接，点击调用知识库部分的SearchServlet，访问对应教程页面。



1. **小结及回顾**
   1. **小结**

整个学期java课程收获颇丰，每次的小作业都要花好几个小时乃至一天去学习，再实践。期末大作业更是促使我们快速成长。感谢老师认真的备课和助教学长批改作业的辛苦，大作业还帮我们求情。

我在该项目中，采用了以下技术和工具：

MVC框架：

View，由html5、CSS3、JavaScript等技术构成

Model：采用MyBatis（实则封装了JDBC），和service构成，MyBatis调用数据库，Service处理逻辑后向Controller层提供服务。

Controller：继承了HttpServlet，获取View层传递过来的数据，根据需求调用Model，实现数据库和前端页面的交互，属于逻辑处理层。

Tomcat服务器运行web项目

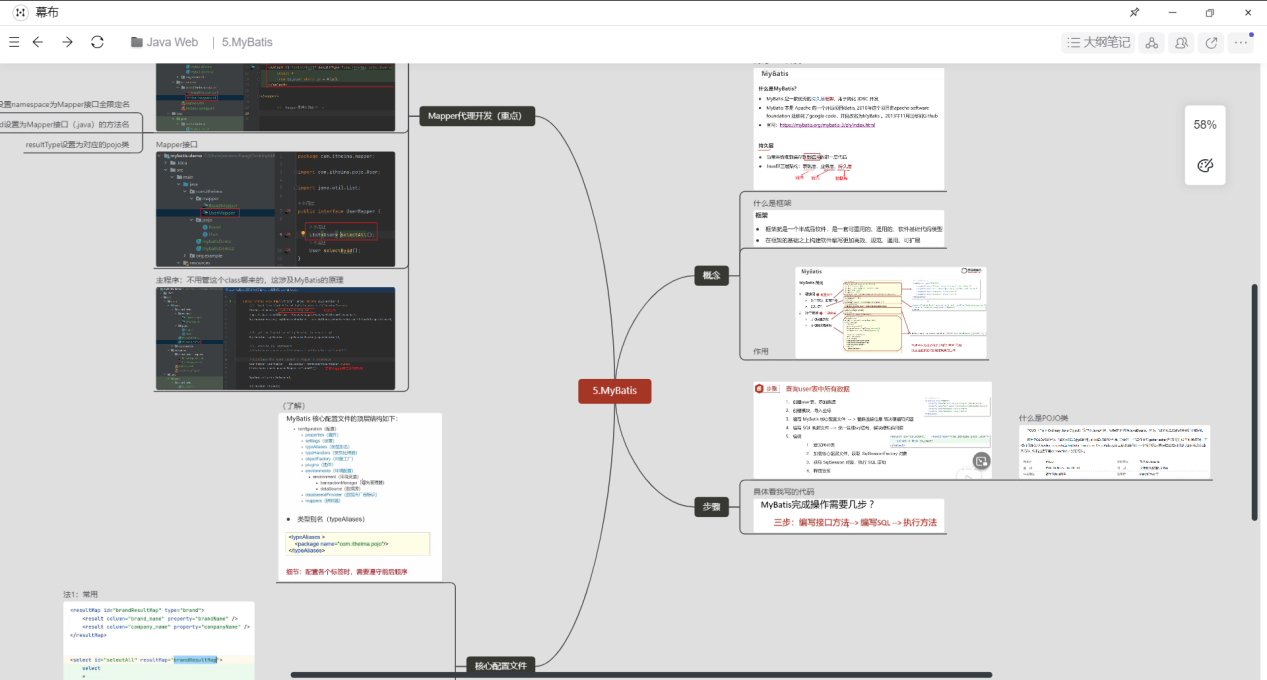
数据库的增删改查、触发器

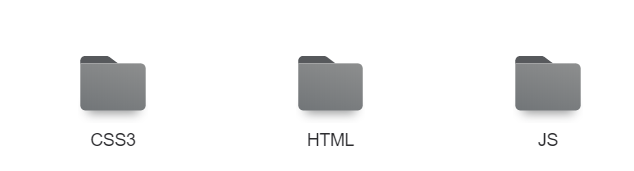
云服务Linux命令学习

* 1. **回顾**

为了把java学得更加的扎实，我在B站上自学了超过100小时的教学视频。记录了以下笔记（太多，只展示了一部分）：







刚开始学的时候有点绝望，因为要学的技术栈实在太多了。而且那时候也不知道大作业是写网页项目，没有练手的机会。后来布置大作业后，快马加鞭，变学边写，学习的效率和乐趣大大提升。