

**本科实验报告**

|  |  |
| --- | --- |
| 课程名称： | 数据库原理 |
| 姓 名： | 王龙龙 |
| 系： | 生仪学院 |
| 专 业： | 测控技术与仪器 |
| 学 号： | 3180101317 |
| 指导教师： | **黄正行** |

2020年 3 月 19 日

**实验报告**

专业：测控

姓名：王龙龙

学号：3180101317

日期：2020.6.19

地点：

装 订 线

课程名称：数据库原理 指导老师： 黄正行 成绩：\_\_\_\_\_\_

实验名称：安装软件以及数据库设计

=================================================================================

1. 实验目的

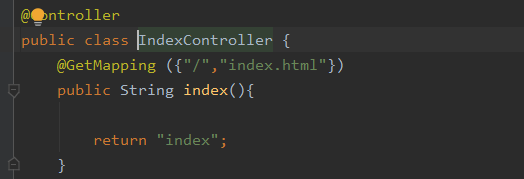
实验报告5：前端涉及页面跳转的剩余功能等的开发，对应第六次实验课(5月14日)；

1. 实验原理和内容

1。编写跳转界面

1. 实验数据记录、处理和分析

1.登录跳转页面：

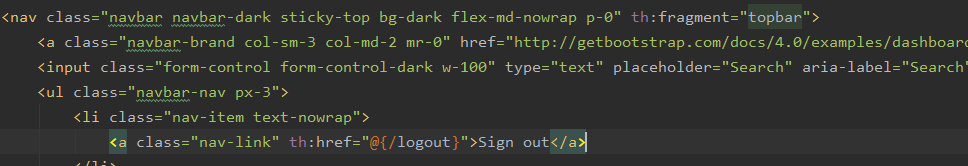


在“/”以及“index.html”请求是进入登录页面：

登录：  

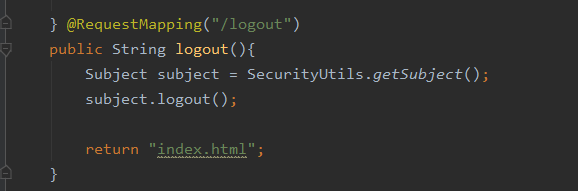

使用shiro接管登录，如果出错则传递相应的报错信息传递给前端，如果登录成功则进入首页

3.顶部导航栏：

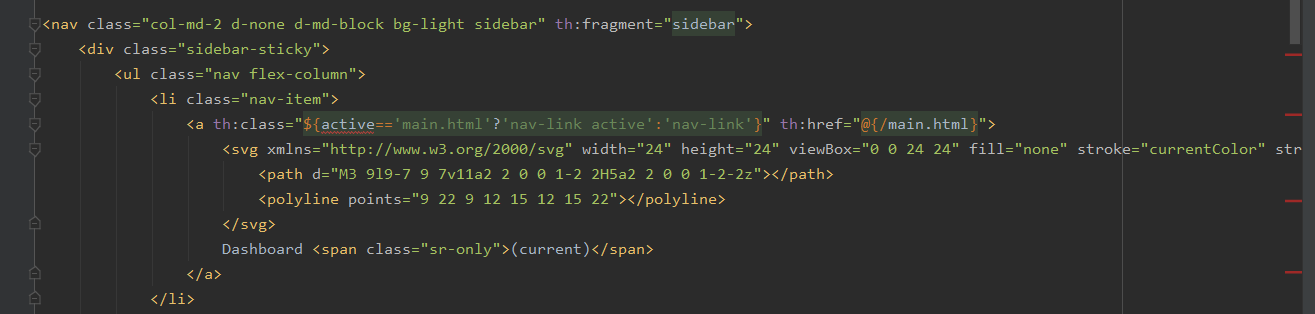


Logout标签对应跳转登出

登出的控制器如下



4.侧面导航栏：



侧面导航栏的：dashboard跳转到main.html,其余都跳转到各自的list页面，如patient跳转到/patienr,而/patient使用的控制器

@RequestMapping("/patient")

return "patient/list";

同时传给patient页面所需要的数据：如所有的patient，以及统计的数据：如不同寿命区间的人数，以及男女数量，健在以及死亡的人数等，

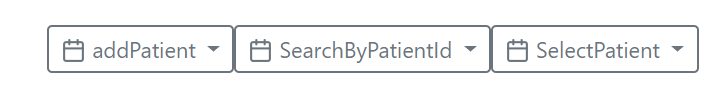
int[] hii=new int[6];  
int[] era=new int[4];  
int male = 0, female = 0, dead = 0, alive = 0;

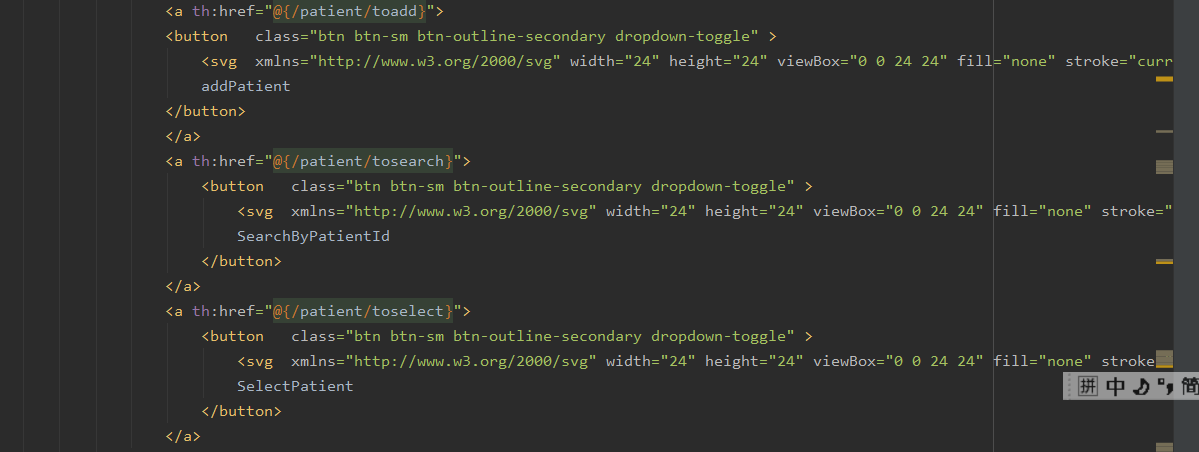
这些数据经过后端提取的，和业务层处理后通过Model传给前端，前端图表处就可以得到相应的图表

List<Patient> patients = patientMapper.queryPatientList();

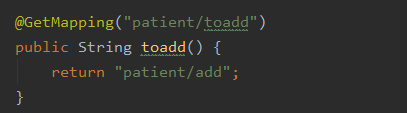
同时，将所有的patient一并交给前端；

顶部三个功能的跳转：

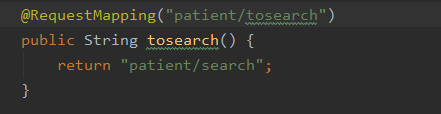




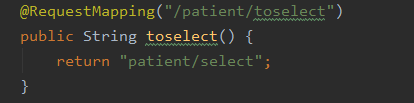
Toadd的请求：  
将用户跳转到add.html



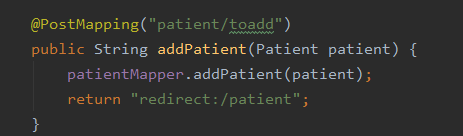
Tosearch请求：跳转到search界面



Toselect请求：跳转到select页面：



添加请求：添加表单提交后：



执行业务代码并重定向至病人首页：  
search和select也是类似，进行相关的业务操作

针对每个病人的按钮操作：



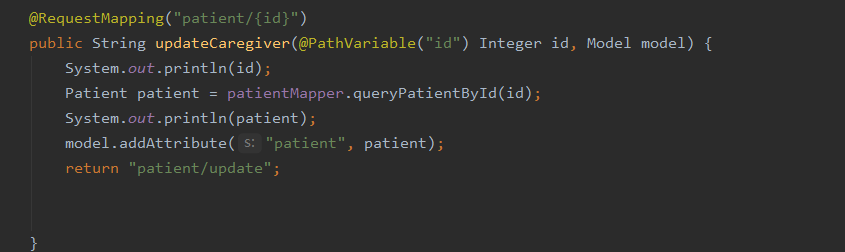
Edit：

改：修改申请：

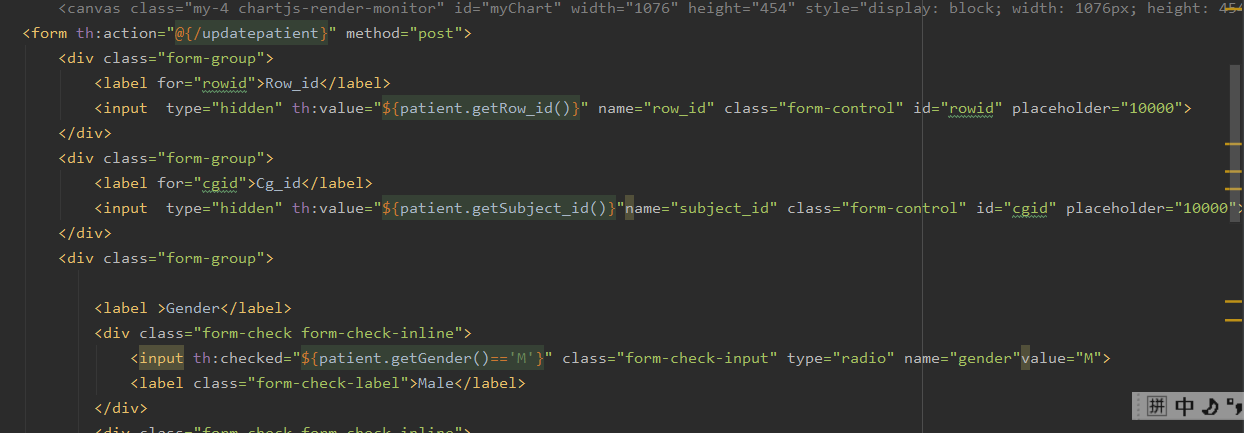
th:href="@{'/patient/'+${patient.getRow\_id()}}

通过url传递row\_id值

对应的controller里接收url里传来的row\_Id,并进行查询回该病人原来就与的信息传回update界面，作为文本框里的默认值

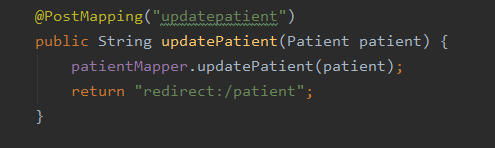


Update界面：



其中表单提交的操作 是updatepatient

对应的跳转控制器：



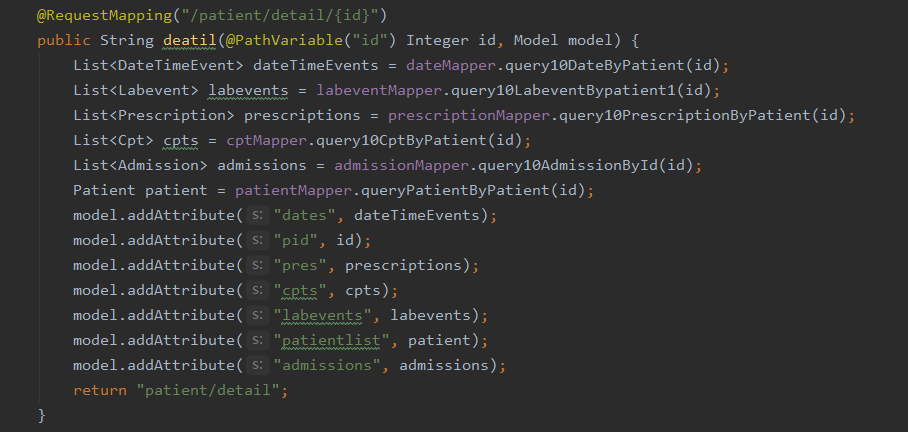
Delete的操作和edit类似，不复赘述

其余几张表的增删查改的链接大致一样；

病人有特制的detailpage；其中用于完成针对该病人的各种操作，和他有关的labevents等表的操作也可以在该页面进行操作

Detailpage:

th:href="@{'/patient/detail/'+${patient.getSubject\_id()}}

对应的视图跳转器：  


调用其他表中所有和该patient有关的信息，一并传入detailpage

在detailpage里有着单独的一些跳转控制器：

如在detailpage里想增加一个admission,使用的控制器和普通的addadmission不同：

1.在detailpage中加入admission应该默认带入该病人的subject\_id,则：

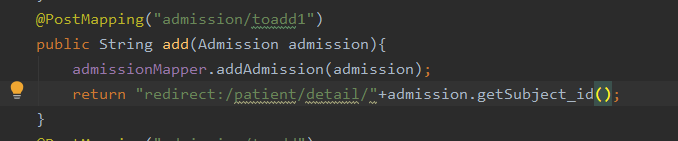
此时addadmission跳转的页面应该是：

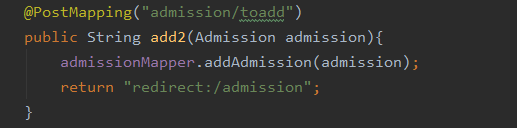
<a th:href="@{'/admission/toadd/'+${id}}">

对应的addpage中，提交表单后重定向的网页也应该不同，从原来对应的重定向到admission首页转为重新重定向到该病人的detailpage

<form th:action="@{/admission/toadd1}" method="post">

对应的跳转控制器：



（原先的跳转：）

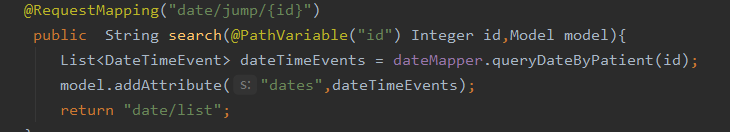
同样的，在detailpage中的所有按钮操作对应的控制器都应该是特制的，操作完成后的重定向操作应该都是该病人的detail页面；

Detail页面中的showall；

Detail中关于病人的其他表的信息都只显示十条，想查看所有可以使用showall,那么可以看到，对特性病人的某张表的showall相当于在该表中进行以subject\_id为关键词的查询：

那么跳转的实现如下

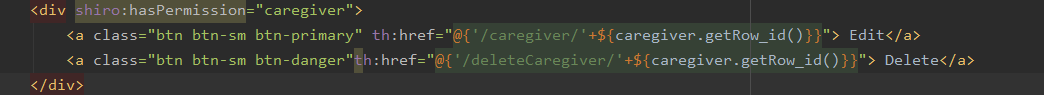
<a th:href="@{'/date/jump/'+${id}}">

对应的controller:  


特定功能的实现：

Caregiver的改删：

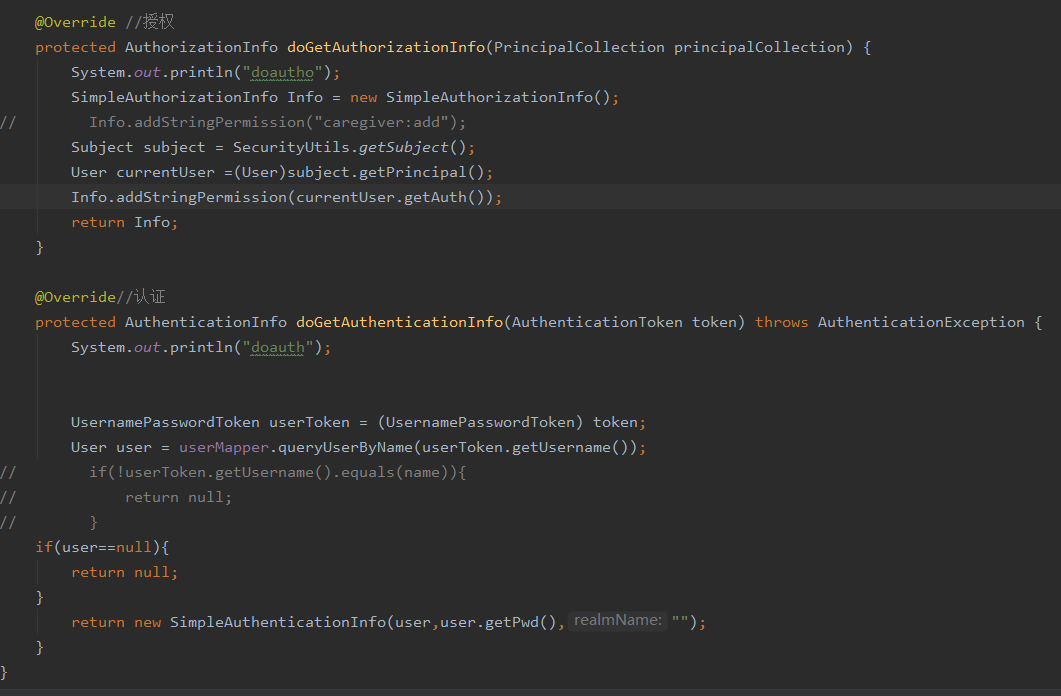
一般的用户应该是没有该权限的，这些按钮不应该出现



使用的方法如上：

shiro中权限的提取是从数据库中自定义的用户表中提取，并在登录事件中赋给该用户

shiro部分授权和认证如下：



还需要设置login界面的url

bean.setLoginUrl("/index.html");

1. 讨论

在detailpage中，我选择的方式是非常粗糙的定制控制器和定制镜像的增删查改表单页面，完成实验后，想到了更好的解决方法，其实只需要定制detaillpage中的每个操作链接就行了，每个链接最后加上subject\_id=?{xxxx.getsubject\_id}，修改原先的控制器使之使用于两种情形，@requestParam 加上patient的subject id，其中将subject\_id改为（required=null）,不一定需要这个参数。在admission页面增加admission时，就不需要传入病人的id，此时判断subject\_id为null,这样重定向到admission的首页，如果接收到了subject\_id，说明是在病人的detail页面传来的请求，那么此时就应该重定向回该病人的detail页面