# 1 项目介绍

## 1.1 项目背景

随着现代互联网的快速发展，人们的生活越来越离不开互联网带给我们的便捷。在这样的环境下，传统的医院医疗系统，现场挂号，就诊，取药等方式已经不能满足我们，所以线上预约挂号，看病的需求迫切要得到解决。在这样市场的要求下，我们的线上医疗系统应运而生。

## 1.2 业务模型

系统框架

前

台

系

统

患

者

模

块

医

生

模

块

后

台

系

统

# 2 项目架构

## 2.1 框架技术

后台框架：Spring Boot

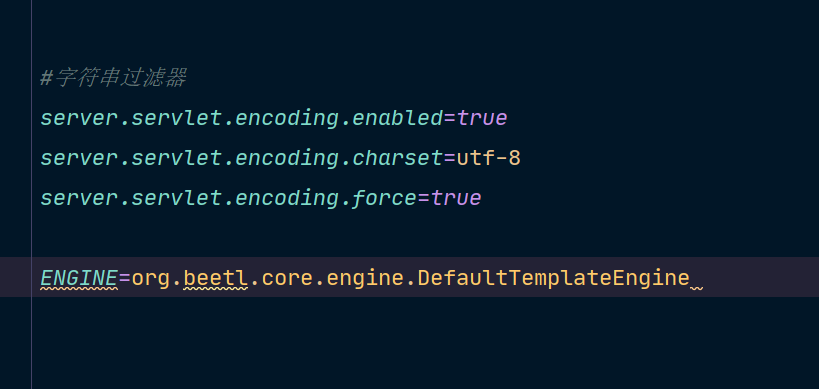
持久层框架： mybatis + mybatis plus

引擎模板：Beetl

## 2.2 maven依赖

## 2.4 yml文件配置

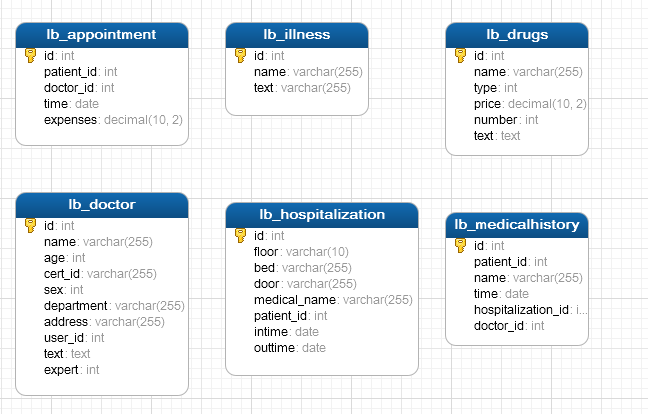


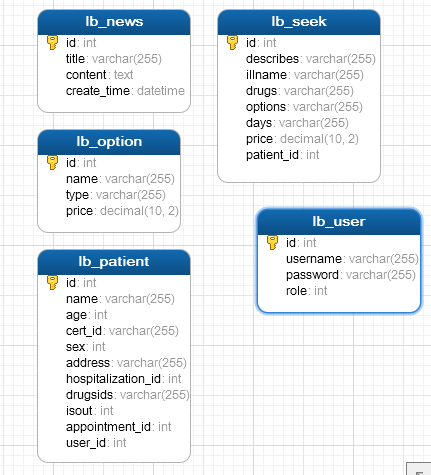


## 2.6 数据库设计

数据库版本：MySQL 5.7.22

表设计：

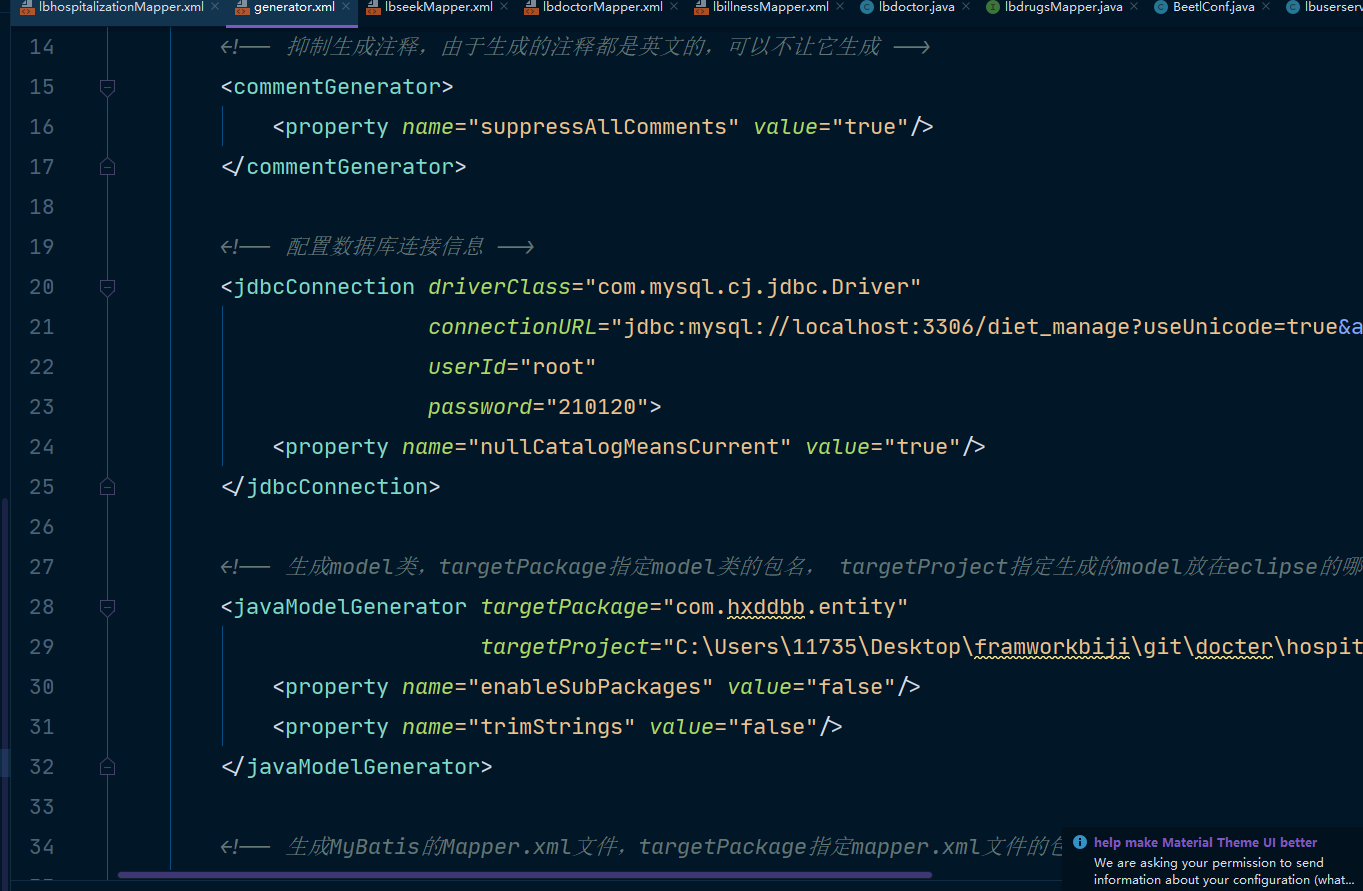




具体的SQL脚本参考项目素材包。

## 2.7 逆向工程

使用generator帮我们将数据表逆向生成pojo、sql文件以及接口。





自动生成器基本配置：

<?*xml version*="1.0" *encoding*="UTF-8"?>  
<!DOCTYPE *generatorConfiguration* PUBLIC "-//mybatis.org//DTD MyBatis Generator Configuration 1.0//EN"  
 "http://mybatis.org/dtd/mybatis-generator-config\_1\_0.dtd">  
  
<generatorConfiguration>  
  
 *<!-- 指定连接数据库的JDBC驱动包所在位置，指定到你本机的完整路径 -->* <classPathEntry *location*="C:\Users\11735\Desktop\framworkbiji\mysql-connector-java-8.0.27.jar"/>  
  
 *<!-- 配置table表信息内容体，targetRuntime指定采用MyBatis3的版本 -->* <context *id*="tables" *targetRuntime*="MyBatis3">  
  
 *<!-- 抑制生成注释，由于生成的注释都是英文的，可以不让它生成 -->* <commentGenerator>  
 <property *name*="suppressAllComments" *value*="true"/>  
 </commentGenerator>  
  
 *<!-- 配置数据库连接信息 -->* <jdbcConnection *driverClass*="com.mysql.cj.jdbc.Driver"  
 *connectionURL*="jdbc:mysql://localhost:3306/springboot?useUnicode=true&amp;characterEncoding=UTF-8&amp;serverTimezone=GMT%2B8"  
 *userId*="root"  
 *password*="210120">  
 </jdbcConnection>  
  
 *<!-- 生成model类，targetPackage指定model类的包名， targetProject指定生成的model放在eclipse的哪个工程下面-->* <javaModelGenerator *targetPackage*="com.hxddbb.model"  
 *targetProject*="C:\Users\11735\IdeaProjects\Springboot\springboot-transaction02\src\main\java">  
 <property *name*="enableSubPackages" *value*="false"/>  
 <property *name*="trimStrings" *value*="false"/>  
 </javaModelGenerator>  
  
 *<!-- 生成MyBatis的Mapper.xml文件，targetPackage指定mapper.xml文件的包名， targetProject指定生成的mapper.xml放在eclipse的哪个工程下面 -->* <sqlMapGenerator  
 *targetPackage*="Mapper"  
 *targetProject*="src/main/resources">  
 <property *name*="enableSubPackages" *value*="false"/>  
 </sqlMapGenerator>  
  
 *<!-- 生成MyBatis的Mapper接口类文件,targetPackage指定Mapper接口类的包名， targetProject指定生成的Mapper接口放在eclipse的哪个工程下面 -->* <javaClientGenerator *type*="XMLMAPPER"  
 *targetPackage*="com.hxddbb.dao"  
 *targetProject*="src/main/java">  
 <property *name*="enableSubPackages" *value*="false"/>  
 </javaClientGenerator>  
  
 *<!-- 数据库表名及对应的Java模型类名 -->* <table *tableName*="student" *domainObjectName*="Student"  
 *enableCountByExample*="false"  
 *enableUpdateByExample*="false"  
 *enableDeleteByExample*="false"  
 *enableSelectByExample*="false"  
 *selectByExampleQueryId*="false"/>  
  
 </context>  
  
</generatorConfiguration>

# 3 后台管理员端

## 3.1 用户登录

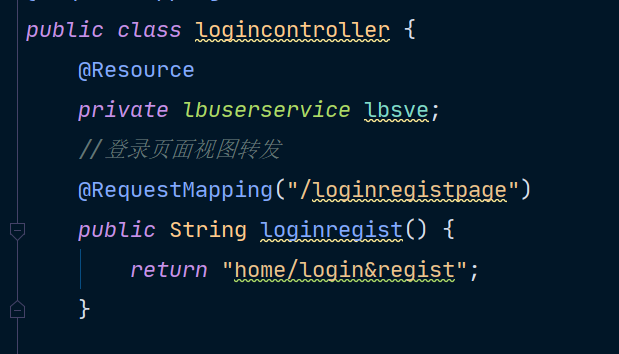
### 3.1.1 添加beetl配置

此处使用配置类的方式实现





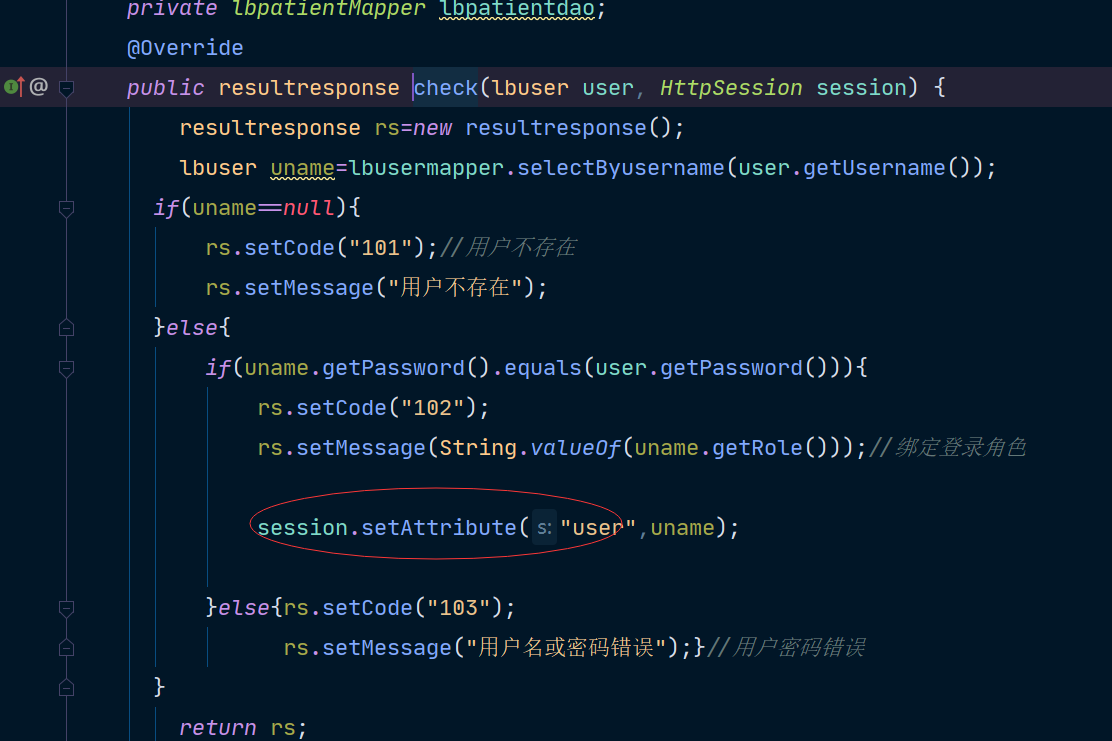
### 3.1.2 定义登录视图控制器



### 3.1.3 开发登录控制器



### 3.1.3 开发service



## 3.2 用户注册

### 3.2.1 注册流程图

注册表单信息

身份证?

注册管理员

查询医生和病人

医生or病人?

注册医生

注册病人

结束

### 3.2.2 开发业务层

这里需要注意，我在判断完用户注册后，由于登录的用户和病人医生等角色不能对接，所以要重新定义接口将用户的主键跟医生病人等外键做连接。同时把注册的id更新到医生表当中的userid字段，只有更新了才能做连接，用户登录就能判断他的角色，因为用户和其他角色是有一定区别，反正需要注意







### 3.2.3 应用路径

在beetl3.0以后的版本中，系统默认提供一个key以便让我们获取系统的应用路径。

查看源码 org.beetl.ext.web.WebRender 57行：

template.binding("ctxPath", request.getContextPath());

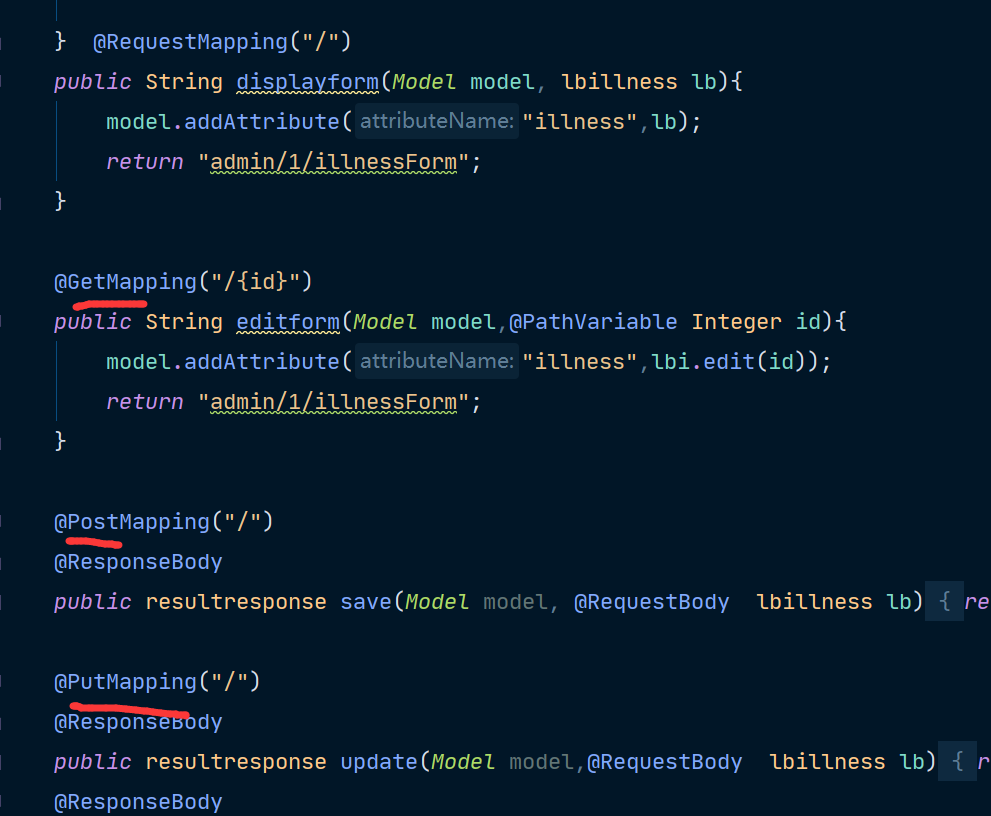
## 3.4 使用REST重构请求地址

没重构之前只能这样判断，我才能不同的需求用一张表单，比如我修改和添加

要用到同一张表单



重构过后



## 3.5 病人管理

病人管理模块列表展示涉及到多表联查。

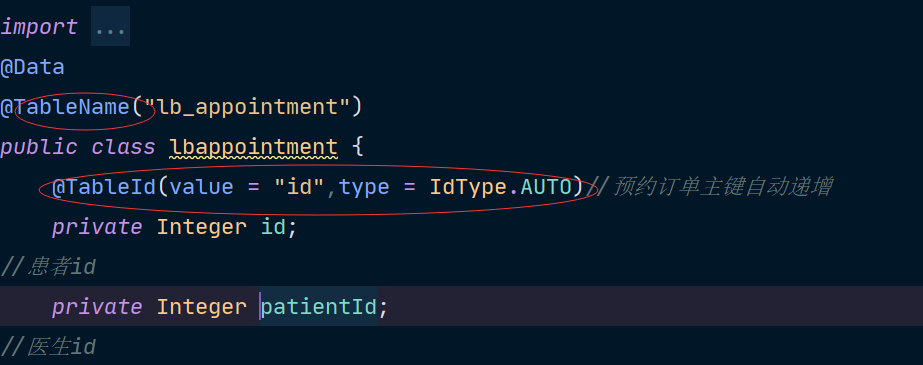
### 3.5.1 定义SQL



注意使用page内置函数可以实现分页。

Mybaits内置函数：

尤其注意 records，需要把函数输入到前端，每个工具的属性名称可能都不同，也有可能是list ---等等，最好自己查阅使用工具文档自行了解，比如还有主键自增注解，表名注解，映射注解，都需要去查阅



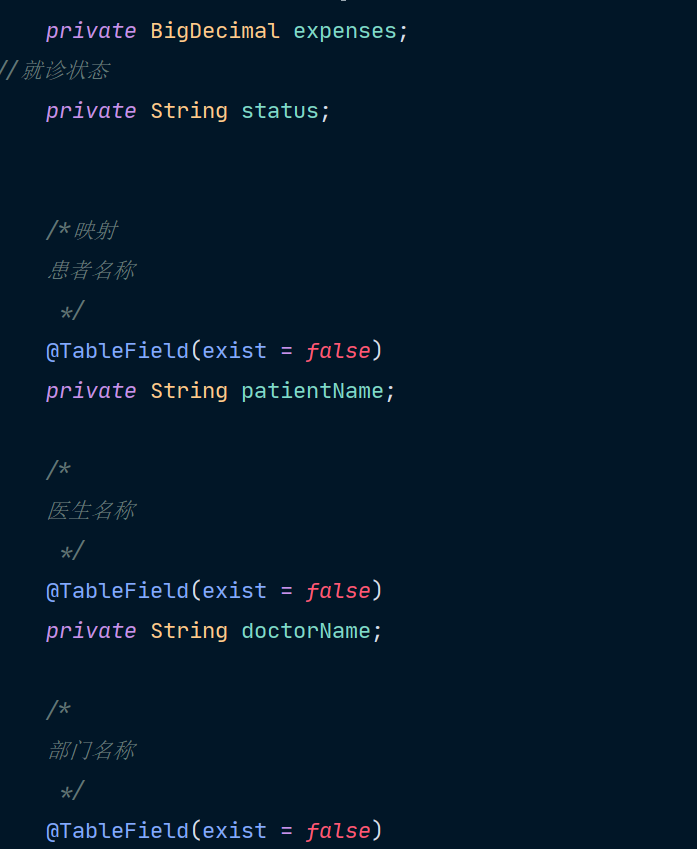
### 

### 3.5.2 开发service

一定注意 querrywrapper条件一定一定！！！不能作为全局变量，否则在二次使用sql语句的时候会和第一次的sql语句进行拼接，自测！



### 3.5.3 实体类中添加额外属性

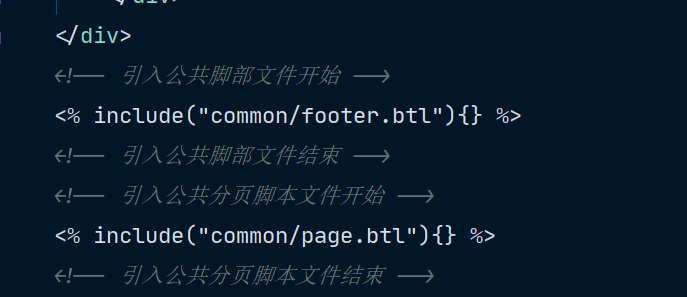


新增医生名称映射字段

## 3.6 抽离分页脚本

### 3.6.1 公共脚本

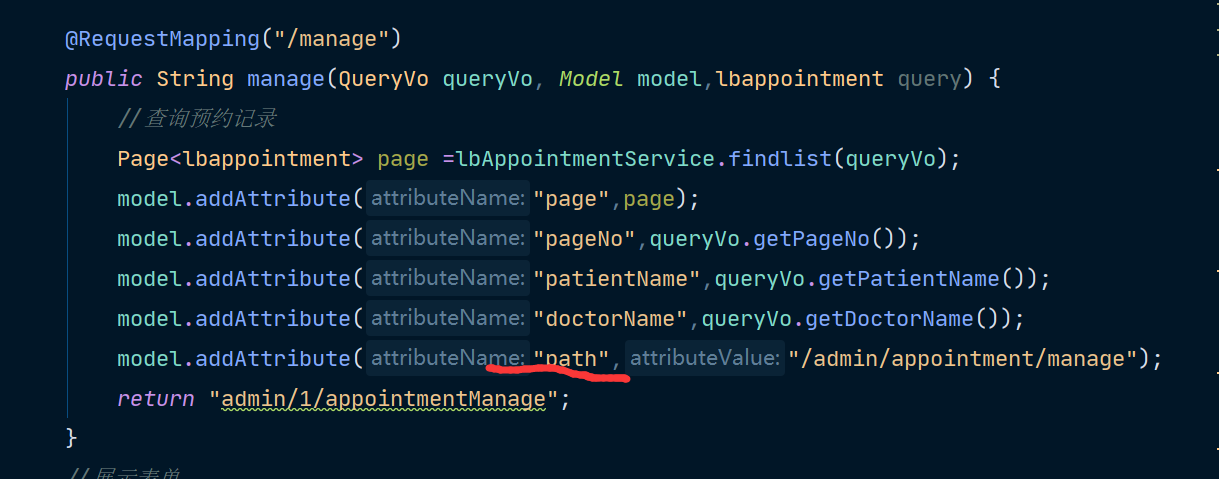
考虑到分页的代码有点冗余，设置一个变量脚本



谁引用，当前就是谁的path



### 3.6.2 后台设置请求路径,不要忘记把Path对象保存到前端



model.addAttribute(**"path"**,**"/admin/patient/manage"**);在model中添加当前方法的请求路径。

## 3.7 药品管理

参考【医生管理】模块开发。



## 3.8 科目管理

参考【医生管理】模块开发。



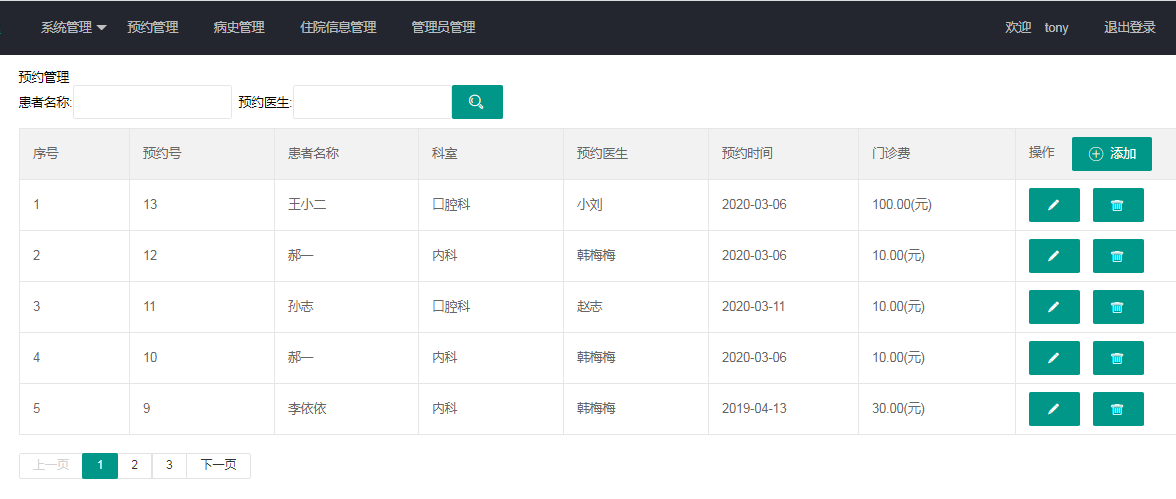
## 3.9 疾病管理

参考【医生管理】模块开发。



## 3.10 预约管理

参考【医生管理】模块开发。



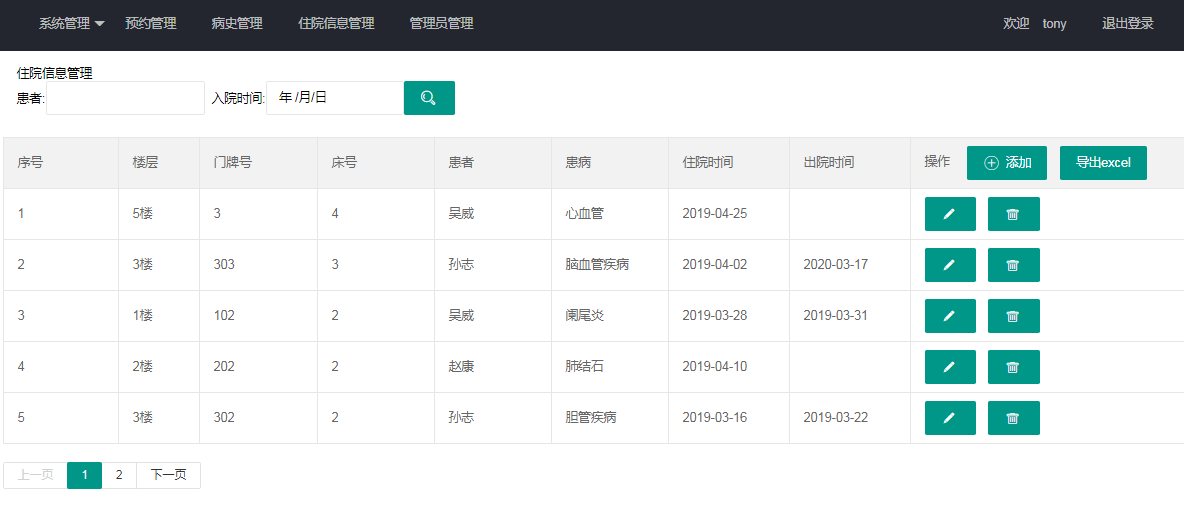
## 3.11 病史管理

参考【病人管理】模块开发。



## 3.12 住院信息管理

参考【医生管理】模块开发。



### 3.12.5 管理员管理

参考【医生管理】模块开发。

## 3.14 登录拦截控制

### 3.14.2 注册拦截器

实现WebMvcConfigurer接口自定义web的配置。



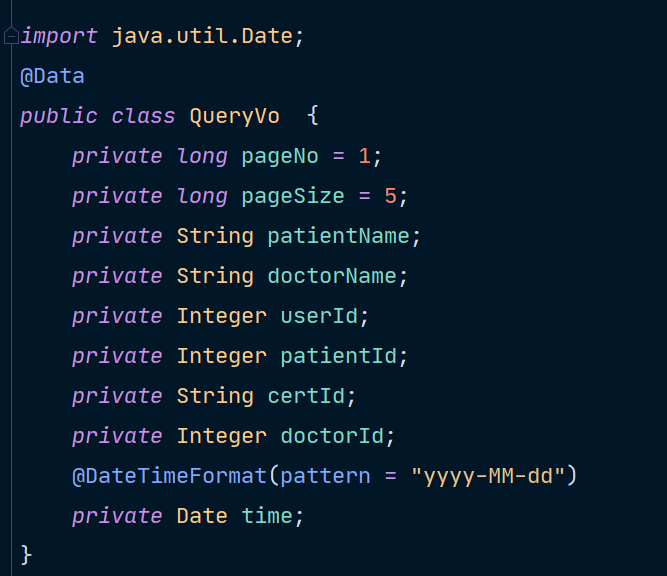


# 4 前台患者端

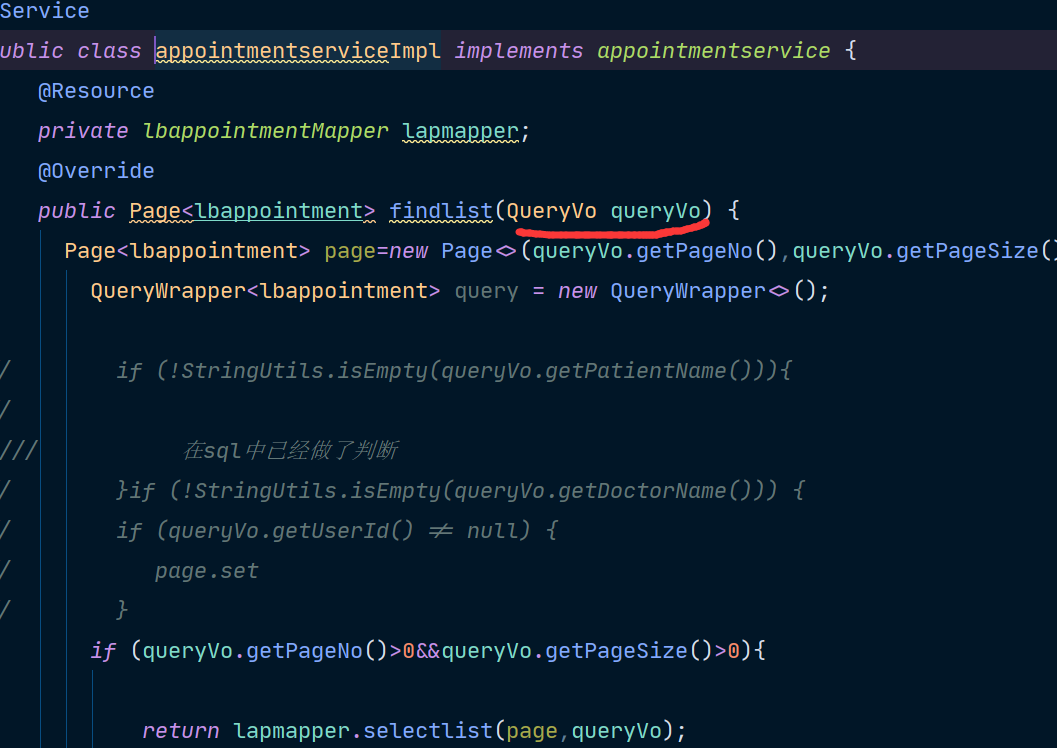
## 4.1 预约功能

### 4.1.1 预约记录

新增查询vo类重构预约查询功能。这个类有 需要查询到的数据相对应的属性



Service接口方法修改为：



### 4.1.2 预约看病

预约信息保存成功，返回插入记录的主键。在BaseMapper中使用insert方法实现。**插入对象返回主键，在mybatisplus里面的insert方法执行完成时已经把主键ID赋值回来给实体ID了，直接get.id**

更新患者的预约记录id，修改业务层的更新方法：





注意：update方法可以只针对不为null的值做更新。

### 4.1.3 生成预约挂号单

使用IText生成PDF。



<dependency>  
 <groupId>com.itextpdf</groupId>  
 <artifactId>itextpdf</artifactId>  
 <version>5.5.13.1</version>  
 </dependency>  
 <dependency>  
 <groupId>com.itextpdf</groupId>  
 <artifactId>itext-asian</artifactId>  
 <version>5.2.0</version>  
 </dependency>  
</dependencies>

## 4.2 信息查询

患者可以在该模块下查询疾病、医生、药品等信息。

界面原型：



# 5 前台医生端

## 5.1 预约患者模块

实现参考其他模块

## 5.2 就诊功能模块

实现参考其他模块

## 5.3 查看患者病史

实现参考其他模块

## 5.4 打印就诊单

实现参考其他模块

## 5.5 重构患者预约记录列表

对于lb\_seek表，添加预约记录主键以区别当前的预约是否已经被医生就诊，添加预约记录状态来确定患者就诊进度（未就诊、就诊正、已就诊）。